A SCHEME FOR IMPROVING THE ANGULAR RESOLUTION IN A VHF RADAR

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

By

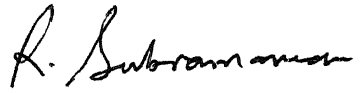
Sqn. Ldr. R. M. NAIR

to the

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
SEPTEMBER, 1979

CERTIFICATE

This is to certify that the work on 'A SCHEME FOR IMPROVING THE ANGULAR RESOLUTION IN A VHF RADAR' by Sqn. Ldr. R.M. Nair has been carried out under my supervision and this has not been submitted elsewhere for a degree.



(R. Subramanian)
Assistant Professor
Department of Electrical Engineering
Indian Institute of Technology
KANPUR.

September, 1979

EE-1979-M-NAI-SCH

ACKNOWLEDGEMENT

It is with a deep sense of gratitude that I express my thanks to my supervisor Dr. R. Subramanian.

To my instructors, I acknowledge my indebtedness for exposing me to so many fascinating subjects.

I thank Shri C.M. Abraham not only for the excellent typing job but also for his perennial cheerfulness over the last two years.

My wife's contribution by making a home for me and my children inspite of the difficulties of the last two years, is beyond any words that I could find.

To my service, the Indian Air Force, I express my everlasting gratitude for having afforded me this wonderful opportunity for study.

R.M. NAIR
Sqn. Ldr.

CONTENTS

ABSTRACT

A brief survey of some of the methods of improving angular resolution of a VHF ground based radar is carried out. The concept of spatial frequency is highlighted. A scheme analogous to holography is proposed using the idea of a DFT on spatial samples to obtain a spatial frequency spectrum. A DFT implementation using table look up instead of a hardware multiplier is suggested and simulated. Elementary simulation of the scheme is carried out.

CHAPTER 1

INTRODUCTION

A ground-based surveillance radar used for the detection of aircraft has to, in addition to detecting the presence of aircraft, be able to determine the position in terms of slant range, azimuth and elevation angles. Such position has to be determined to the greatest possible accuracy. In addition to locating the position accurately it should also be possible to differentiate between two targets proximate to each other. The closer the proximity the better the resolution. Better accuracy does not necessarily mean better resolution nor vice-versa.

In this thesis we are concerned only with obtaining improved angular resolution in the azimuth plane. If corresponding resolution is not available in the elevation dimension also, the system will not be useful for positioning friendly aircraft in an advantageous position vis-a-vis hostile aircraft. It could of course be helpful in determining the number of hostile aircraft, unless they were 'stacked' one above the other. As a navigation aid to friendly aircraft its usefulness would of course be enhanced, since aircraft normally know their attitude, their range and azimuth angle from a reference point can be passed to them with a better accuracy.

The normal means of achieving this has been to use higher radiated frequencies so that for given mechanical load (for aerial rotation) and structural complexity narrower beams and hence better resolution is achieved. In this thesis we consider a system using a radiated frequency in the VHF range. Use of VHF frequency confers two advantages :

a) freedom from weather clutter and 'angels' [1]. At 200 MHz the returns from rain is 60 dB lower as compared to that at 10 GHz.

b) for a given pulse repitition frequency (p.r.f) a lower transmitted frequency (larger wavelength) gives a higher blind speed. As will be seen later for the case under consideration, the blind speed is far above the normal operating speeds of aircraft.

The thesis is organized in the following manner. Chapter 2 deals with some methods which have been proposed/used for improving angular resolution. The chapter after dealing with optical holography attempts to draw a picture of an analogous system in the microwave region. This leads, in Chapter 3, to specifications of the solution to the problem at hand, based on the holographic analogy. The chapter spells out the details of the solution. Chapter 4 discusses some of the routines used in the simulation as well as consideration of SNR and clutter to signal ratios. Chapter 5 gives the conclusions arrived at as well as some suggestions.

CHAPTER 2

METHODS FOR IMPROVING ANGULAR RESOLUTION

Before proceeding to details of the proposed scheme, a brief review of some of the methods used for obtaining improved angular resolution is presented below.

The methods considered are :

a) Superdirective arrays
b) Interferometric methods
c) Frequency swept imaging
d) Synthetic aperture methods
e) Microwave holography

2.1 SUPERDIRECTIVE ARRAYS

A superdirective antenna is one having a beamwidth (in radians) $<<$ 1/largest dimension of the array in wavelengths [2]. A characteristic feature being their dependence on heavy cancellation between closely spaced radiators. Thus requiring a high precision in feed and placement of the elements. Thus not only the achievable gain of such arrays is to be considered, in addition one must keep in view the sensitivity of the lobe pattern to errors in feed and placement. As mentioned in [2] these arrays are high Q small

bandwidth devices, as the amount of reactive energy in the near field is very large compared to the energy radiated per cycle. Schelkunoff [3] has also shown that though directive gain can be increased the efficiency and bandwidth reduces.

Newman et al [4] quote results obtained for a 9 element linear broadside array of copper halfwave dipoles with overall array length of $\lambda/4$ for obtaining a directivity 8.5 times that of a single dipole. The efficiency was 10^{-14} percent and accuracy required was one part in 10^{11}. They have however, shown that for a receiver array where the system performance is limited by external noise the poor efficiency does not effect the signal to noise ratio (provided (antenna efficiency x external noise) $>>$ internal + receiver noise). Such a system cannot obviously be used in the situation under consideration.

2.2 INTERFEROMETRIC METHODS

By the term 'interferometric' methods we mean those methods where the antenna aelements are spaced apart far enough to give grating lobes, i.e. if the output from the elements were additively combined. We consider two methods.

2.2.1 1st Method

This method was proposed by M.I. Skolnik [5]. The method is based on recognizing the change in grating lobe pattern

that results from a change in λ /d ratio (d is element spacing). He considers a radar covering a volume of space with one broad transmitting beam and many fixed narrow receiving beams. The receiving array and transmitting antenna are seperate. The receiving array consists of 11 isotropic elements spaced 5.75 apart, giving a main lobe of 0.9°. The grating lobes are at $\pm 10^{\circ}$, $\pm 20.4^{\circ}$, $\pm 31.5^{\circ}$, $\pm 44.1^{\circ}$, $\pm 60.5^{\circ}$. The region between the grating lobes is covered by 9 additional beams generated by beam forming networks. By transmitting on another frequency (or alternately using another array with different d) the position of the grating lobes can be made to shift while the main beam remains in the same position. By taking the new d/λ ratio as 5.2 a shift of one beamwidth in the 1st grating lobe is produced (the 2nd grating lobe will shift by twice, the 3rd by three times and so on). Thus ambiguities can be resolved. This method appears to be suited for the problem under consideration. Using two sets of antenna with different inter-element spacing as above one could expect a feasible solution. In the presence of a number of targets and clutter this might be difficult.

2.2.2 2nd Method

This method is the multiplicative or correlation array. A two-element correlation array is shown in Fig. 1.

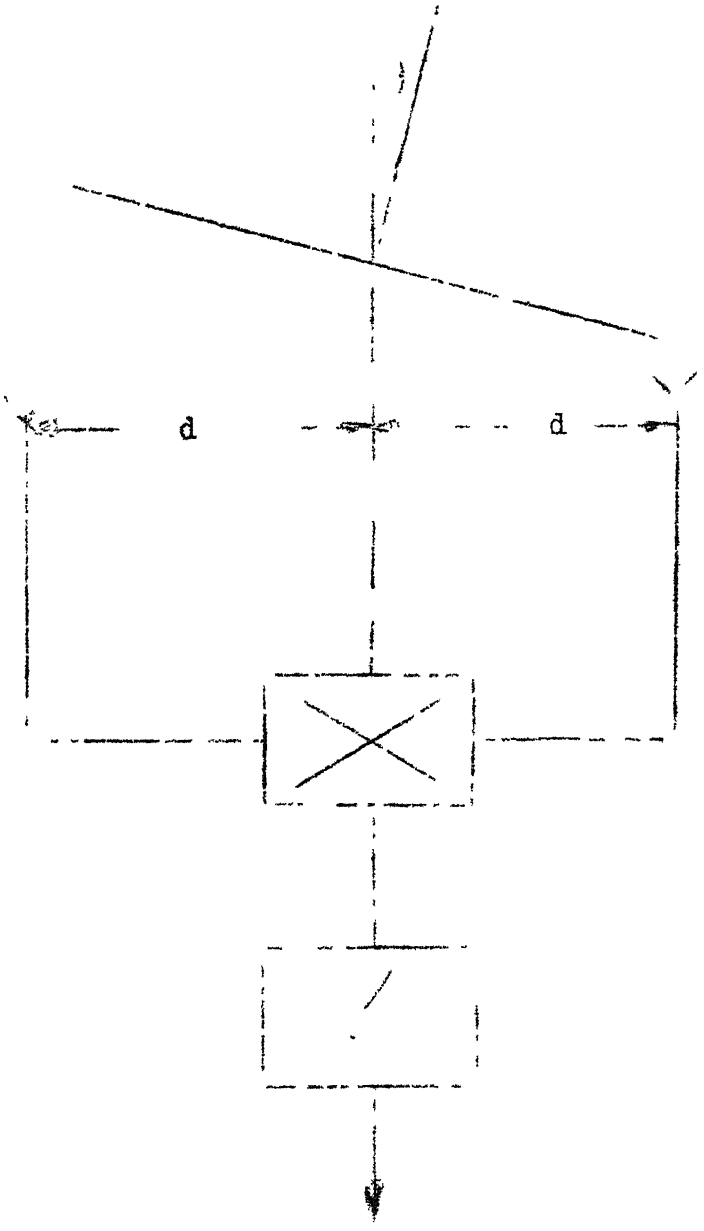


Fig. 1

The output is a function of the relative delays (between the output of the antenna element) and hence of the angle of arrival. The output $D_{11}(\tau)$ is given by

$$D_{11}(\tau) = \frac{1}{2T} \int_{-T}^{T} V(t)\ V(t+\tau) \cdot dt$$

where

$\tau = 2nd \sin \theta / C$

$2T$ = Time period of received sinusoidal waveform.

The above expression closely resembles the autocorrelation function $\rho_{11}(\tau)$

$$\rho_{11}(\tau) = \lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} V(t)\ V(t+\tau)\ dt$$

In most cases $D_{11}(\tau)$ is proportional to $\rho_{11}(\tau)$.

$D_{11}(\tau)$ represents the directional response of the array (as τ is a function of θ). Since $D_{11}(\tau)$ is also proportional to the autocorrelation of the received waveform it gives the fourier transform of power spectral density of the received waveform. Thus, if we use a waveform with a autocorrelation equal to $\delta(0)$ (e.g. white noise) the angular response is similar to a delta function at angle $\theta = 0$. If we used a pure sinusoid the directional response would exhibit grating lobes.

The above is a manifestation of the space frequency equivalence [6]. This equivalence concept indicates that two widely spaced receivers of small size can achieve a directivity against a multifrequency source equal to the directivity obtained from a fully extended linear array of receivers of twice the length against a single frequency source. It must be noted that no mention is made of the received signal to noise ratio.

Figure 2 taken from KOCKS paper illustrates this equivalence.

McCartney [7] concludes that as with many processing systems employing nonlinear devices the signal/noise performance is poorer at low SNR. Thus in radar it would mean reduced range. However, the range resolution is also poor. This is because pulse widths have to be large enough so that for the lowest modulating frequency two relatively delayed signals will always have an integral number of cycles of overlap. Further though the directional response is narrow the presence of other targets affects it thus the resolution is poor unless the returns from the second object is uncorrelated to the first objects response - a condition which cannot be expected in a radar.

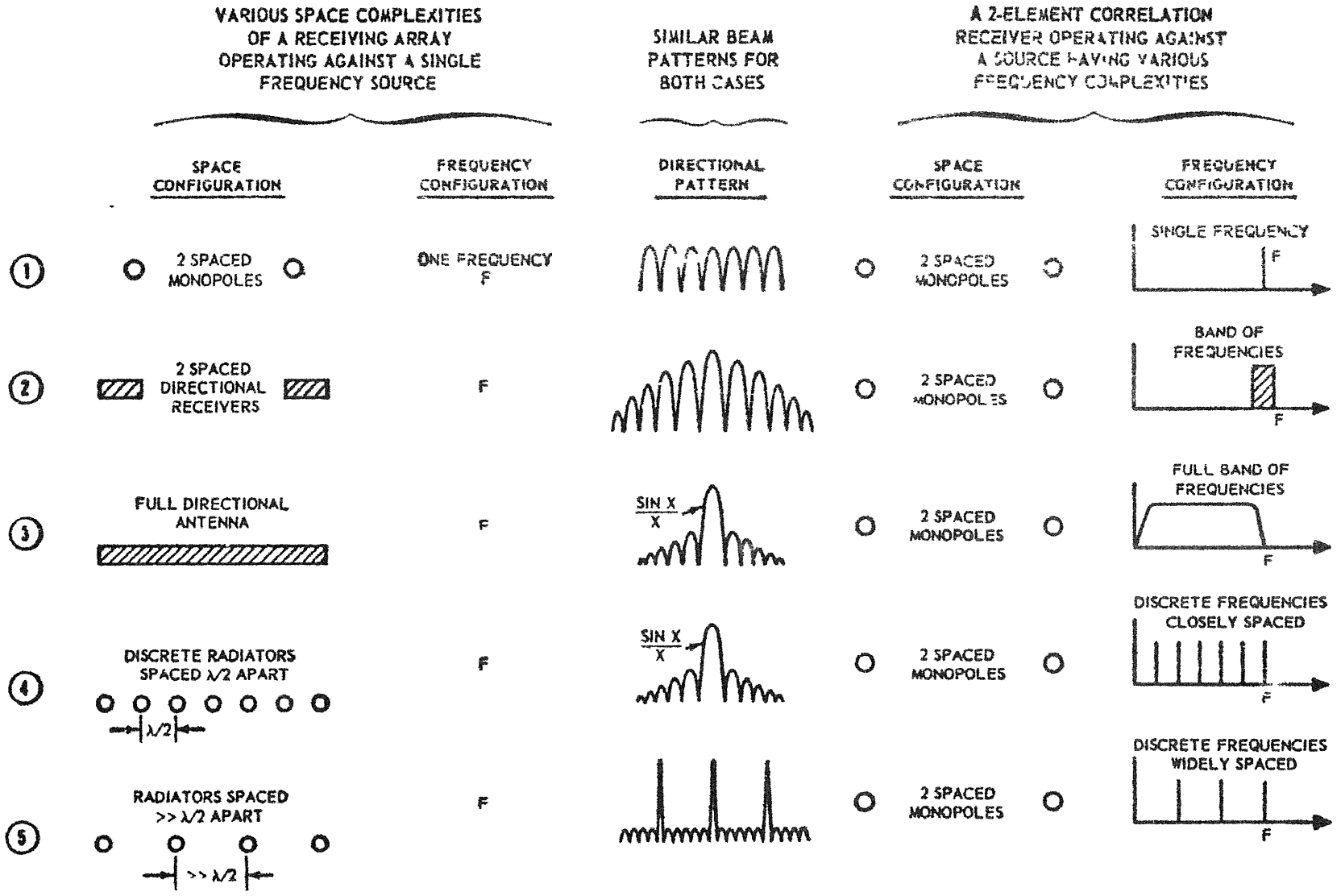


Fig. 27—An equivalence exists in a receiving array between the complexity of its space configuration and the complexity of the frequency configuration of the source

FIG – 2

2.3 FREQUENCY SWEPT IMAGING

Proposed by Farhat [8]. As stated by him - 'An essential part of any coherent imaging system is an imaging aperture that furnishes a spatial sample of the wavefield scattered by a coherently illuminated object'. Images are then formed from the sampled data through basically a fourier or Fresnel transform operation.

Two coherent wavefronts will interfere and cause interference patterns just as in the optical case. The concept of frequency swept imaging is based on the 'breathing' (expansion or contraction) of the pattern when the illuminating frequency is changed.

Consider Fig. 3. O1 and O2 are scattered plane waves of unit amplitude. R is the reference plane wave with amplitude $A >> 1$. The time varying component of the receiver output at B is

$$I(t) = 2A[\cos \{k_1 B(\sin \theta_{O1} - \sin \theta_{OR})(1 + \alpha t)\} + \cos \{k_1 B(\sin \theta_{O2} - \sin \theta_{OR})(1 + \alpha t)\}] \qquad (2.1)$$

where a linear frequency sweep $w(t) = w_1(1+\alpha t)$ of duration T is used where w_1 is the initial angular frequency. This may also be written as

$$\lambda(t) = \lambda_1/(1+\alpha t) \qquad (2.2)$$

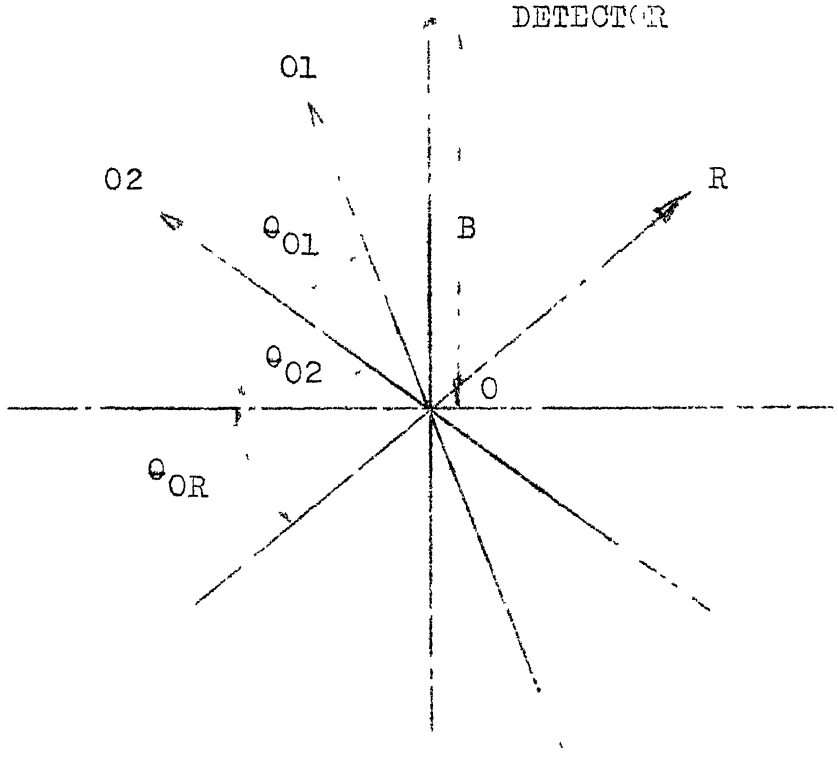


Fig. 3

if λ_2 is the terminal wavelength, then,

$$\alpha T = (\lambda_1/\lambda_2 - 1) \qquad (2.3)$$

The above expression can be put as

$$I(t) = 2A[(\frac{e^{ja_1(1+\alpha t)} + e^{-ja_1(1+\alpha t)}}{2}) + \frac{e^{ja_2(1+\alpha t)} + e^{-ja_2(1+\alpha t)}}{2}]$$

where $a_{1,2} = k_1 (\sin \theta_{01,2} + \sin \theta_{0R})B$

Considering the fourier transform of one term $e^{ja_1(1+\alpha t)}$

$$F(w) = \int [e^{ja_1} \cdot e^{ja_1\alpha t}] \cdot e^{-jwt} \cdot dt$$

$$= e^{ja_1} \cdot \delta(w - a_1\alpha)$$

However, since t can only take values such that $0 \leq t \leq T$. The above transform has to be convolved with $[T\,\mathrm{sinc}(wT/2)]$.

We, therefore, have $F(w) = e^{ja_1} T\,\mathrm{sinc}[(w - a_1\alpha)\, T/2]$

Similar expressions can be obtained for the other terms.

Therefore the fourier transform of I(t) is

$$I(w) = AT[e^{ja_1}\,\mathrm{sinc}\,[(w-a_1\alpha)\, T/2 + e^{-ja_1}\,\mathrm{sinc}\,(w+a_1\alpha)T/2$$

$$+ e^{ja_2}\,\mathrm{sinc}\,(w-a_2\alpha)T/2 + e^{-ja_2}\,\mathrm{sinc}\,(w+a_2\alpha)T/2\,] \qquad (2.4)$$

This now represents the image reconstruction along with the conjugate image parts. By the Rayleigh resolution criterion the minimum resolution is obtained for

$$a_1\alpha - a_2\alpha = \frac{2\pi}{T} + \frac{2\pi}{T}$$

Therefore, $\alpha(a_1 - a_2) = \frac{4\pi}{T}$

$$\alpha(k_1 \sin \theta_{01} - k_1 \sin \theta_{02})B = \frac{4\pi}{T}$$

Substituting for α

$$\frac{(\lambda_1/\lambda_2 - 1)}{T} (\sin \theta_{01} - \sin \theta_{02})\, k_1 B = \frac{4\pi}{T}$$

Substituting $k_1 = 2\pi/\lambda_1$,

$$\sin \theta_{01} - \sin \theta_{02} = \left[\frac{1 \times 2}{\lambda_1/\lambda_2 - 1}\right] \cdot \frac{1}{B} \qquad (2.5)$$

If we had a sweep of 2 - 18 GHz and a B = 1 km the resolution for small values of θ_{01} and θ_{02} is worked out below.

$$\lambda_1 = 150 \times 10^{-3} \text{ m} \qquad \lambda_2 = 1350 \times 10^{-3} \text{ m.}$$

$$\theta = \theta_{01} - \theta_{02} = \left[\frac{150 \times 10^{-3} \times 2}{\frac{1}{9} - 1}\right] \times \frac{10^{-3}}{1}$$

$$150 \times 10^{-6} \times 2 \text{ radian}$$

This is a resolution of 30 metres at 100 kms.

This method again is not suitable because it involves wideband transmitting systems.

2.4 SYNTHETIC APERTURE

In this method a large aperture is synthesized by sequentially transmitting (and receiving) from different points. This in effect amounts to time sharing a single transmitter/receiver by the elements of an array. The returns from each point are stored, weighted and added. The principle has been used for airborne synthetic aperture radars used for terrain mapping. Synthetic apertures are also used in radio astronomy. In this the motion of the earth provides the necessary translation of the receiver. An example of such a system is the Ooty Radio Telescope working at 327 MHz [9].

Harmuth [10,11] discusses stationary synthetic aperture radars based on nonsinusoidal functions.

The similarity between the coherent processing of the airborne SAR and holography has often been highlighted [12,13,14,15]. As pointed out by Kock [14] a feature which sets holography and coherent sidelooking radar apart from ordinary photography and radar is that it is not just one plane that is in focus. This is because each point forms its own zone plate. Since formation

of holograms do not require a doppler shift such doppler free coherent radars should also be possible. 'They could take the form of long linear arrays of independent receivers, crossed arrays (Mills Cross) or square arrays'.

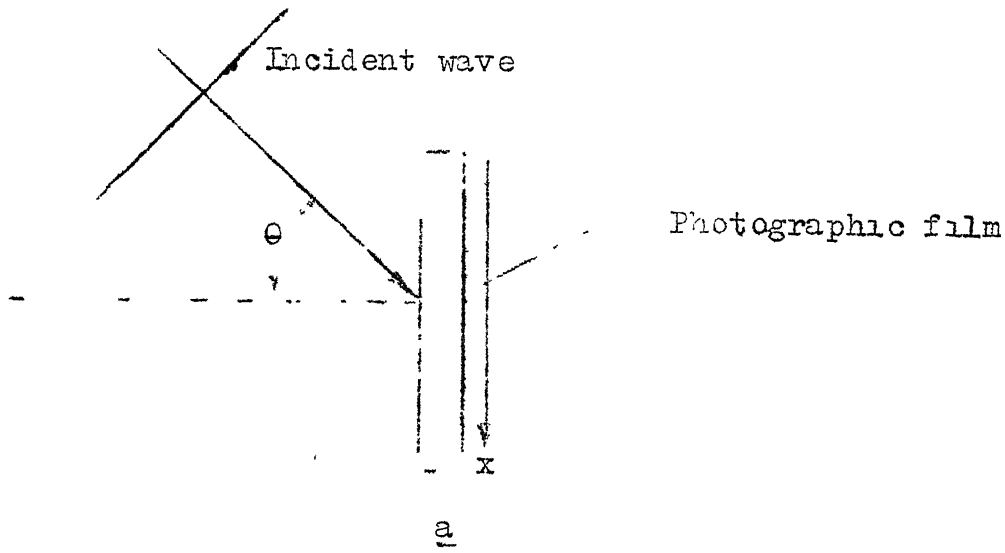
We now turn to Microwave Holography.

2.5 MICROWAVE HOLOGRAPHY

In conventional photography the photographic film records the light intensity over the exposed portion. The incident light however has an associated phase. The variation of phase (which depends on the wavelength and the angle of arrival) over the film however is not recorded. This variation of phase over the film, if the illumination is coherent, alongwith the intensity variation provide the total information of the incident light wave. It may therefore be used to reconstitute the incident wavefront.

By introducing another coherent wavefront called a reference wave, we obtain an interference pattern which reflects the phase information. Consider Fig. 4.

In Fig. 4(a) we have only illumination by the coherent incident wave of wavelength λ of amplitude A_1 and a phase variation along x of $\frac{2\pi}{\lambda} \times \sin\theta$. (We consider only the one dimensional case which can readily be extended to two dimensions).


Incident wave
θ
Photographic film
x
a

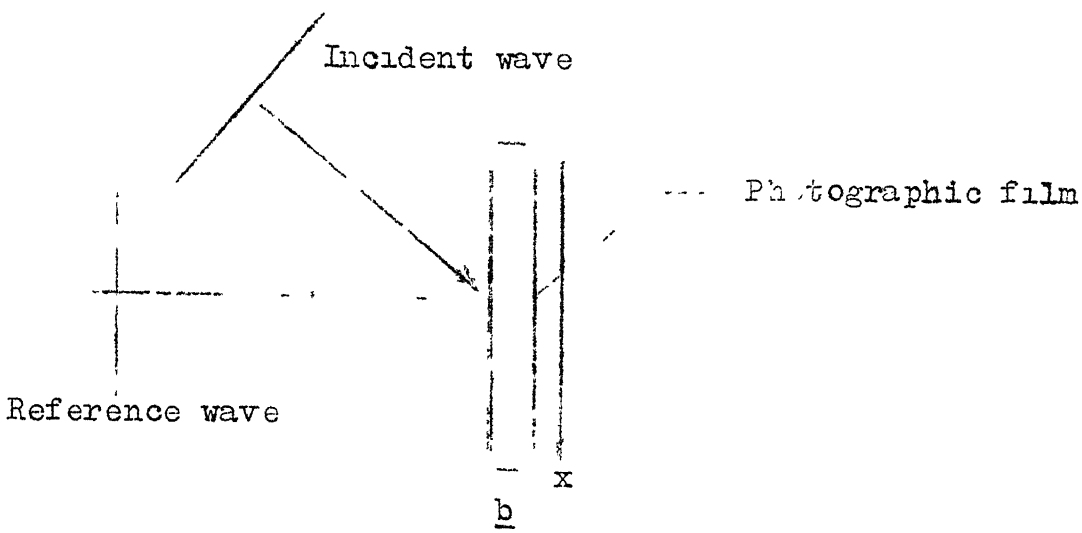

Incident wave
Photographic film
Reference wave
x
b

Fig. 4

The film will therefore have a uniform exposure proportional to the intensity A_i^2

In Fig. 4(b) a coherent reference wave has been introduced. The reference wave since it is normal to the film has a constant phase along x. The reference wave has unit amplitude.

The two wavefronts will now produce maxima and minima depending upon where they are in phase and where in antiphase. In the above case the maximas will be seperated by

$$x = \lambda / \sin \theta$$

Thus the variation of intensity over the film records the phase information. The amplitude of incident wave determines the amplitude of the variation i.e, the intensity of the maximas and minimas. Since only the real part will be recorded the film will record a pattern given by

$$A_1 \cos \left(\frac{2\pi x \sin \theta}{\lambda}\right) \qquad (26)$$

With reference to Fig. 4(a) we have mentioned that the phase (φ) as a function of x is given by (considering only the x dependent part of the intensity).

$(2\pi/\lambda)\ x \sin \theta$ (taking $\varphi = 0$ at $x = 0$) (2.6a)

i.e. $d\varphi = (2\pi/\lambda) \sin \theta\, dx$

$$\cdot \frac{1}{2\pi} \frac{d\varphi}{dx} = \sin \theta / \lambda$$

The left hand side is termed the spatial frequency (f_x). Where the temporal frequency has units of cycles per unit time, spatial frequency has units of cycles for unit distance.

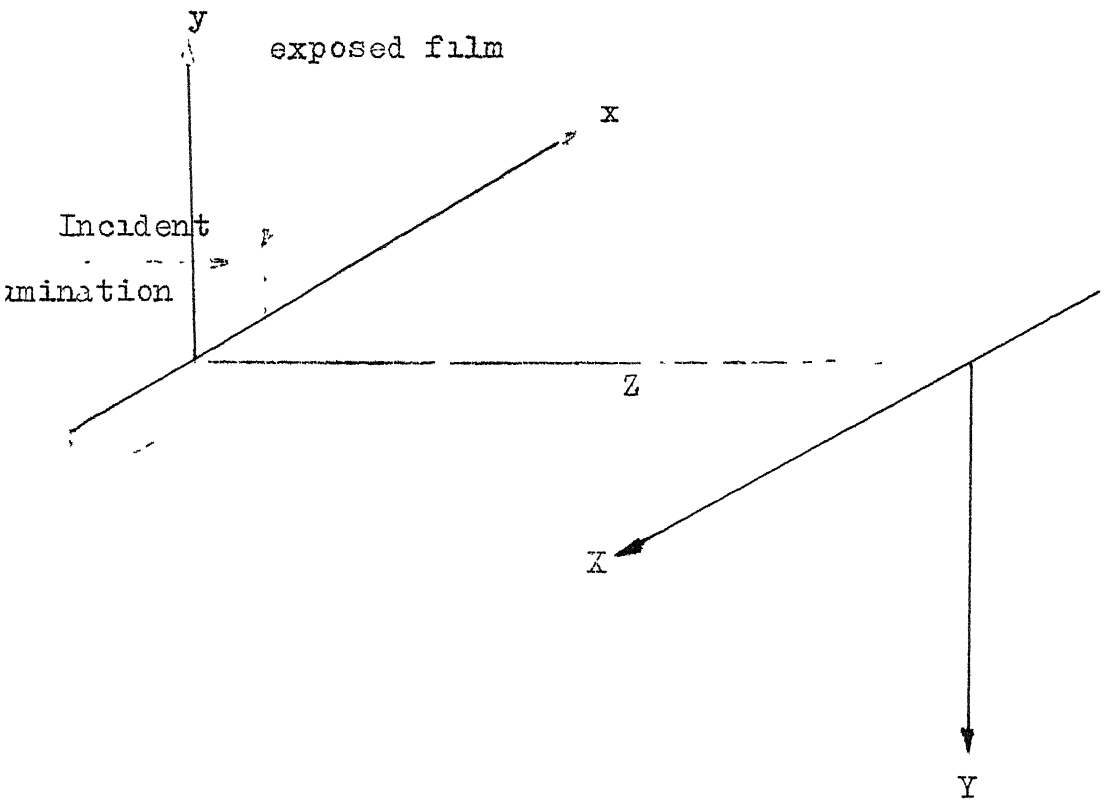
We see that

$$f_x = \sin\theta/\lambda = 1/\text{distance between interference fringes}$$

Now if we took $-\theta$ as the angle of incidence the spatial frequency would be $-\sin\theta/\lambda$ but the spacing between the fringes wo would still be the same. We thus have an ambiguity in the phase information 'coded' in the fringe pattern. It is easy to see that this arises from the fact that only intensities are recorded. Thus the phasor which leads the reference phasor by α will produce the same resultant amplitude as a phasor which lags by α.

~~If~~ Now the exposed film is illuminated by a coherent plane wave and the ~~intensities~~ amplitudes observed in a plane parallel to the plane of the film and situated at a distance Z from it as shown in the Fig. 5.

An elemental area in the xy plane will transmit amplitude $A_i(x,y)\,dx\cdot dy$. When observed from point X in the XY plane this results in an amplitude $[A_1(x,y)\,dx\cdot dy]/Z$ (assuming $Z^2 \simeq (x+X)^2 + (y+Y)^2$). The associated phase change is $(2\pi/\lambda)\sqrt{(X+x)^2 + (Y+y)^2 + Z^2}$. The resultant at X of all such elemental areas is given ~~by~~ below (the discussion does not consider the part of the exposure independent of (x,y)). The amplitude



Note : the direction of X and Y are reversed w.r.t. x and y

Fig. 5

transmittance is proportional to the exposure [16].

$$A_o(X,Y) = \iint \frac{A_i(x,y)}{Z} \exp[j(2\pi/\lambda)\sqrt{(x+X)^2+(Y+y)^2+Z^2}]\cdot dx\,dy \quad (2.7)$$

Simplifying

$$\sqrt{(X+x)^2 + (Y+y)^2 + Z^2} = Z\sqrt{1 + (\frac{X+x}{Z})^2 + (\frac{Y+y}{Z})^2}$$

Taking the first terms of the binomial expansion :

$$Z[1+\tfrac{1}{2}\ (\frac{X+x}{Z})^2 + \tfrac{1}{2}\ (\frac{Y+y}{Z})^2] \qquad \text{Fresnel Approximation}$$

$$= Z + \frac{1}{2Z}\ [(X+x)^2 + (Y+y)^2]$$

$$A_o(X,Y) = \iint \frac{A_1(x,y)}{Z} \exp[-j\ \frac{2\pi}{\lambda}\cdot Z]\ \exp[-j\ \frac{2\pi}{\lambda}\ \frac{1}{2Z}$$

$$\cdot [\ (X+x)^2 + (Y+y)^2]]\ dx\ dy \qquad (2.8)$$

$$= \frac{\exp[-j\ \frac{2\pi}{\lambda}\ Z]\cdot \exp[-j\ \frac{2\pi}{\lambda}\ \frac{1}{2Z}(X^2+Y^2)]}{Z} \iint A_1(x,y)\exp[-j\ \frac{2\pi}{\lambda}\ \frac{1}{2Z}(x^2+y^2)]$$

$$\cdot \exp[-j\ \frac{2\pi}{\lambda}\ \frac{1}{Z}\ (xX+yY)]\cdot dx\ dy \qquad (2.9)$$

making the Fraunhofers approximation $Z \gg x^2 + y^2$

$$\frac{\exp[-j\ \frac{2\pi}{\lambda Z}\ \{Z^2+\frac{1}{2}(X^2+Y^2)\}]}{Z} \iint A_1(x,y)\ \exp[-j\ \frac{2\pi}{\lambda}[x(\frac{X}{Z}) + y(\frac{Y}{Z})]]\ dx\ dy \qquad (2.10)$$

We put $f_X = X/(Z\ )$, $\qquad f_Y = Y/(Z\ )$

Since the film had recorded the intensity as given in eqn. 2.6 the intensity of the light transmitted will be given by

$$A_i(x,y) = A_i \cos\left(2\pi \frac{x \sin\theta}{\lambda}\right) + B \qquad (2.11)$$

where B is the constant background illumination.
Therefore, the term under the integral sign now becomes,

$$\int A_1 \cos(2\pi f_x x) \exp[-j\, 2\pi f_X x]\, dx \cdot \int \exp[-j\, 2\pi f_Y y]\, dy$$
$$+ B \int\int \exp[-j\, 2\pi f_X x - j\, 2\pi f_Y y]\, dy\, dx \qquad (2.12)$$

If we took the limits of the integration as $\pm\infty$ we see that both terms are fourier transforms. Using the fourier transform relation for a sinusoid and a constant we get the result as

$$[\delta(f_X - f_x) + \delta(f_X + f_x)] \cdot \delta(f_Y) + B\delta(f_X) \cdot \delta(f_Y) \qquad (2.13)$$

Thus we would have got three bright spots on the X-axis. One at the the origin due to the background illumination . The other two at

$$f_X = \pm f_x = (\pm \sin\theta)/\lambda \qquad (2.14)$$

or $$X = \pm Z \sin\theta/\lambda$$

However, because of the Fraunhofer approximation we cannot take the limits of integration as $\pm\infty$ (also because of the finite size of the film). To account for this we can as usual multiply $A_i(x,y)$ with a rectangular window function corresponding to the x-dimension of the film (as we are considering only the one dimensional case). Thus the delta functions are

convolved with a sinc function. Because of this, along with the bright spots at X = 0, $\pm$ Z sin θ, we would get secondary bright spots caused by the sinc function side lobes.

Above we have seen a non-dimensional case of the concept of spatial frequency and how the recording of the fringe pattern and subsequent reconstruction are intimately related to it as also the cause of ambiguity in reconstruction.

However, in general the illumination will be complex with amplitude and phase variation over the surface. Since the film recording will depend only upon the real part, i.e.
$A_i(x,y) + A_i^*(x,y)$

Substituting this for $A_i \cos\left(\frac{2\pi x \sin\theta}{\lambda}\right)$ in eqn. (2.12).

We have for the first term

$$\int [A_1(x,y) + A_i^*(x,y)] \exp[-j2\pi f_X x]dx \int \exp[-j2\pi f_Y y]dy \qquad (2.15)$$

If the reference illumination is changed to an angle α the reference itself would have a phase variation which would produce an equal change in the phase of $A_1(x,y)$ which would now be

$$A_i(x,y) \exp(j2\pi f_\alpha x)$$

The above integral now become

$$\int \exp[-j2\pi f_Y y]dy \cdot \int [A_i(x)\ \exp[-j2\pi x(f_X - f_\alpha)$$

$$+ A_i^*(x,y)\ \exp[-j2\pi x[f_X + f_\alpha]]\ dx \qquad (2.16)$$

Here if $A_i(x,y)$ were caused by a diverging beam-$A_i^*(x,y)$ would represent a converging wavefront. Further the spatial spectrum on the X-axis is now seperated. The virtual image is about $X = +f_\alpha$ and the real image about $X = -f_\alpha$.

The above is a one-dimensional form of the well-known Leith-Upatniek hologram.

It must however be remembered that the fourier transform relation between the amplitude distribution along X and the film exposure along x is valid only if the Fraunhofer condition is satisfied. We have also seen how under this condition the optical reconstruction is equivalent to a fourier transformation. The above is a simple one dimensional formulation of the well known result, that in the Fraunhofer zone the field distribution is a fourier transforn of the aperture distribution [16].

The fourier transformation is between space domain (x,y) and spatial frequency domain (f_X, f_Y). The above discussion is as applicable at radio and acoustic frequencies as at optical. We have seen that the spatial frequency represents the angle of arrival of a wavefront at the plane of aobservation. Therefore, their spectrum will lie between $\pm 1/\lambda$. Therefore, the maximum required spatial sampling rate is $4/\lambda$ samples/unit distance.

However one practical difficulty at radio frequencies compared to optical frequencies is the large size of the holographic 'plate' further corresponding to a photographic plate no good area detectors exist for radio frequencies. Experiments done at 35 MHz using liquid crystal area detectors required power levels at the detector of the order of a few milliwatts/cm^2 [17]. Such high signal level would be considered a radiation hazard! Aircraft returns cannot be expected to be of this order.

CHAPTER 3

STATEMENT OF THE PROBLEM AND SUGGESTED SOLUTION

The problem taken up for study is the improvement of angular resolution for a ground based radar operating in the vhf range (100 - 300 Mc/s) and used for the detection of aircraft.

Some of the representative parameters of the radar set are given below.

a	Peak transmitted power	750 KW
b	Pulse width	6 μs
c	Transmitted frequency	150 MHz
d	Max detection range on (on 1 m^2 target)	350 Kms
e	Pulse repitition frequency	400 pps (giving an unambiguous range of 375 Kms)
f	Antenna scan rate	6 rpm
g	Transmitted beamwidth	15° in azimuth 60° in elevation

Further antenna size increase for improvement of resolution is not considered feasible.

The Scheme Envisaged :

The transmitter system of the available radar is retained. The receiver system however would consist of one or more antenna arrays and receiver processors situated at a distance of

500 - 1000 metres from the transmitter site. Some of the operational advantages that could result from such a scheme are :

a) The transmitter and its antenna system need only be a cheap 'bare-bones' affair thus easily replaced if destroyed. Thus reducing system vulnerability to anti-radiation missiles.

b) By giving operational cover at lower frequencies also it will force a potential jammer to expand his energy over a wider spectrum.

c) As the transmitter and its antenna could be relatively simpler and cheaper deployment of decoys would be much easier.

The price to be paid for all this will be increased processing complexity and a sophisticated data communication system to link the receiver sites.

Approach to a Solution

From what we have seen in Chapter 2, we now make the following observation.

a) Our problem is then to determine the spatial frequency spectrum as this will give us the different directions of arrival.

b) The spatial spectral bandwidth can have a maximum possible value of $2/\lambda$, i.e.[(cos $(-\pi/2)/\lambda$ to cos $(\pi/2)/\lambda$]

c) By invoking the sampling theorem it should be possible to sample only at discrete spatial points and be able to reconstruct the original spatial information.

d) As is well known a convenient way to do this would be by doing a DFT using the FFT algorithm.

A number of experiments have been done using the holographic techniques and then sampling the interference pattern [27,28]

The method of forming interference patterns to determine the phase though invaluable at optical frequencies is not quite appropriate at radio and acoustic frequencies as well established methods for coherent detection already exist [18]. We need not, therefore, have recourse to physically generating interference. However, the notion of discrete spatial samples of both the amplitude and phase of the spatial information and the notion of a spectral analysis/filtering of this discrete sequence could possibly be used.

We are forming a beam by processing. By appropriate filtering the 'beam' may be 'pointed' in any direction at any time after the spatial samples have been recorded in much the same manner as after a holographic record has been made we may

view the scene by illuminating the hologram. The view depending upon the position occupied by the viewer. In our case now the DFT corresponds to the illumination and the position of the viewer corresponds to the frequency we are interested in.

At this point, it would not be out of place to mention a type of frequency scanning technique known as within the pulse scanning. In this system the relative phase shifts of the array elements are varied by rapidly varying the phase shifts of the LO frequency fed to each receiver element [19] . One of the systems reported [20] for scanning is given in Fig. 6 . But, as no record is kept of the waveform received at each receiver, on phase shifting and combining, information from directions other than where the 'processed' beam points is lost. Because of this sampling caused by within the pulse scanning only a fraction of the energy returned from a target is utilized. This loss of signal energy will be a disadvantage.

In the proposed system however the scanning is not done at RF or IF but after coherent detection.

To summarize the above, : It is proposed that the system use a array, the outputs of all elements of the array be coherently demodulated to obtain a complex quantity (the In-phase and Quadrature voltages). Analogous to the holographic method a DFT be done on this sequence of complex samples to obtain the discrete spatial frequency spectrum.

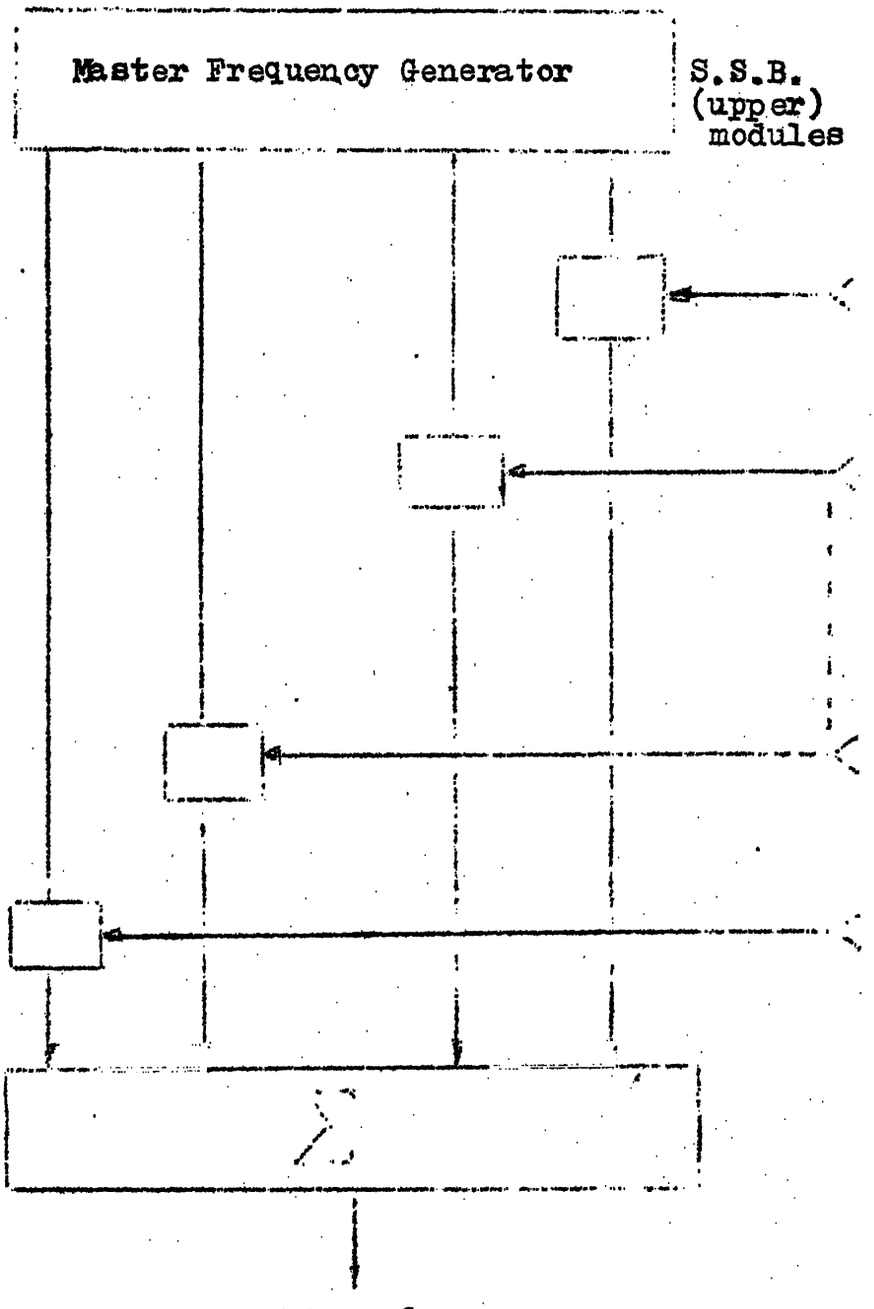


Fig. 6

We now turn to the design of such a system.

3.3 Details

We have seen above that the spatial frequency bandwidth being $2/\lambda$ the minimum sampling rate would have to be $4/\lambda$ samples/unit distance. It is here that we take advantage of the bandlimiting done by the transmitter beam. Because the transmitter beam does not extend from $-\pi/2$ to $\pi/2$, but has a beamwidth of 15°, as mentioned, we will have a spatial spectrum corresponding to this 15° only.

As the rate of change of $\sin\theta$ (w.r.t. θ) is maximum when $\theta = 0$ the maximum spatial bandwidth will be obtained when the beam lies from $(-7.5^{\circ}$ to $+ 7.5^{\circ})$. The spatial bandwidth now will be $0.26/\lambda$ requiring a sampling rate of $0.52/\lambda$ samples/unit distance.

The assumptions in the above are :

a) The transmitter sidelobes are neglected

b) The targets of interest lie at ranges greater than 30 Kms. Thus with a seperation of 0.5 Kms between the transmitter and receiver systems the angular co-ordinate to a target from either point is same.

If we remove assumption (b) all it would mean would be that the locus of equal delay, instead of being a circle, is

an ellipse with the two sites as the focii.

We have now obtained the requirement that the antenna should be seperated by a distance of $\lambda/0.52$ (sampling rate $0.52/\lambda$ sample/unit distance).

Now we would like to know how many of such antennae are required. The more number of samples that we obtain, the better the resolution. As the number of such samples is finite, consider it as applying a rectangular window to an infinite sequence. The infinite spatial sequence if transformed would give a delta function in the spatial frequency domain (this is the ideal case). The rectangular window however, convolves a sinc function with this. The transform will now be a sinc function given by

$$D \; \frac{\sin(2\pi f_X D/2)}{(2\pi f_X D/2)}$$

where the spatial samples have been taken over $\pm D/2$. f_X is the spatial frequency. The first nulls will be at

$$2\pi f_X D/2 = \pm \pi \quad [\text{or } 2\pi(f_X - f_o) D/2 = \pm \pi \text{ if we consider resolution about some other frequency } f_o]$$

$$f_X = \pm 1/D$$

$$\sin \theta' = \pm \lambda/D$$

$$\theta' = \pm \sin^{-1}(\lambda/D)$$

Thus for two objects to be resolved we use the Rayleigh criterion that they must be seperated by at least $2\theta'$.

If we took $\theta' = 1^{\circ}$, i.e., two targets seperated by 2° should be resolved then,

$$D = 57.2\lambda$$

The number of spatial samples $= 57.2\lambda \times \frac{0.52}{\lambda} \simeq 30$ samples

Since we propose to do a DFT on these samples we take the number of samples as 32. Thus the number of antenna used is 32.

Now for 32 samples we determine what would be the resolution at an angle of 0° and at say 60° (since the resolution will get poorer as we approach the endfire position of the array).

at broadside

$$\theta' = \sin^{-1}[\lambda/(32x\lambda/0.52)] = \sin^{-1}[0.0162]$$
$$= 0.93^{\circ}$$

Therefore, resolution $= 1.86^{\circ}$

At 60°

$$[\sin\theta_1 - \sin 60^{\circ}] = \pm\lambda/(32x\lambda/0.52) = \pm 0.0162$$

$$\sin\theta_1 = \pm 0.0162 + 0.865$$

$$\theta_1 = \sin^{-1}(0.8488), \quad \sin^{-1}(0.8812)$$

$$\simeq 58.3^{\circ}, \quad 62^{\circ}$$

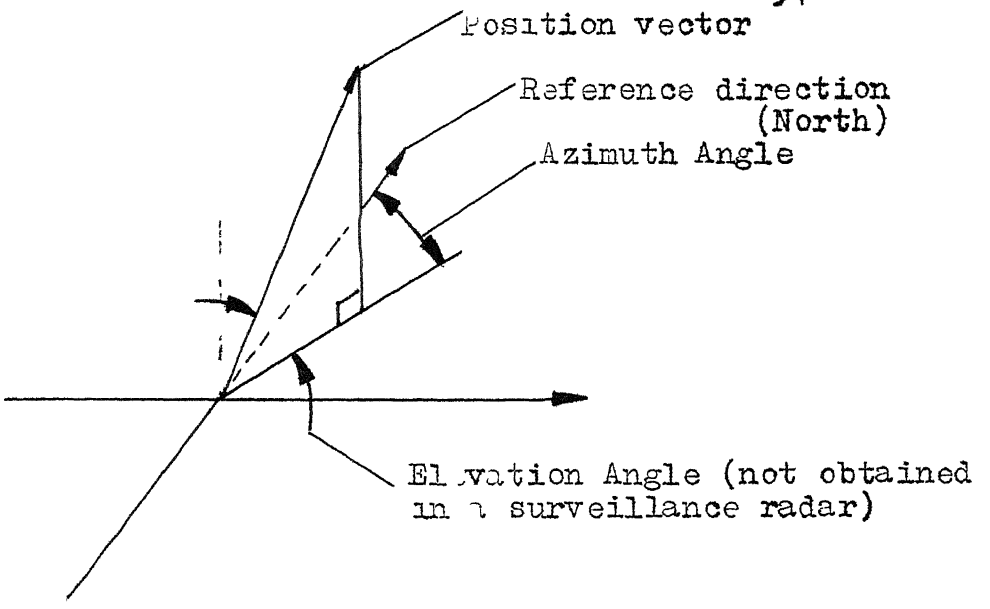
The beamwidth/2 is approximately 1.8° which gives a resolution of approximately 3.6°.

Similarly, at 45° the beamwidth/2 is approximately 1.3° giving a resolution of 2.6°.

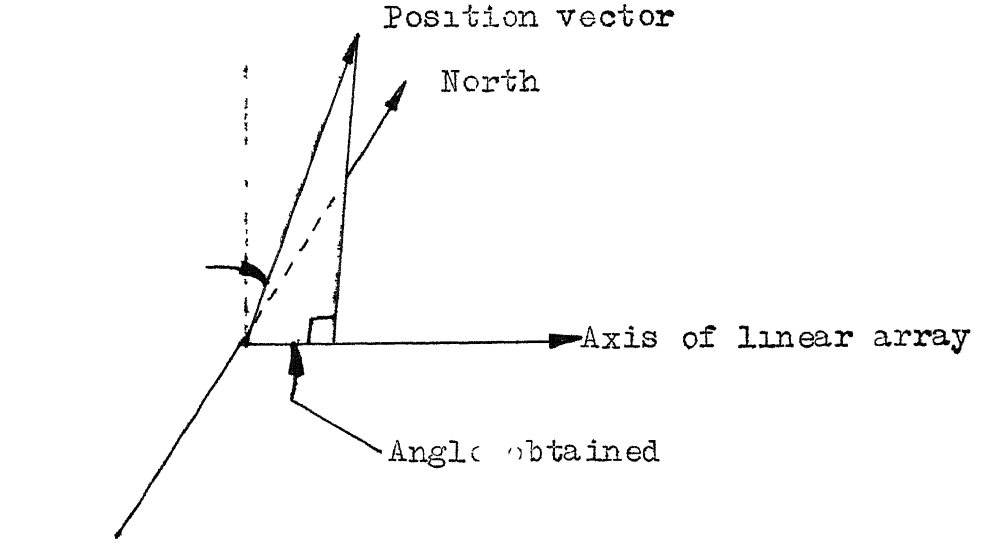
Thus we see that even at 60° we have obtained a satisfactory reduction of the transmitted beamwidth of 15°.

In a normal surveillance radar two targets at the same range and azimuth angle but different heights cannot be resolved. In the system that we are considering, targets which lie at the same range and lying on the surface of a cone with the linear array as its axis will not be resolved. In other words, whereas in a conventional system the co-ordinate system consists of the length of the position vector (range), and the angle of the position vector projected onto the horizontal plane, from a vertical plane passing through the reference direction (North) this is the azimuth angle - in the present system the angle of the position vector from the linear array axis is what is obtained. This is shown in Fig. 7.

As the transmitter beam rotates in azimuth from North (000) clockwise to East (090), South (180) and West (270) what is the corresponding behaviour of the angle output at the receiver system after processing?



Normal surveillance coordinates



Coordinates in proposed scheme

Fig. 7

The DFT will give us 32 spatial frequencies over a spectrum of 0 to $0.52/\lambda$. Are the different sectors corresponding to this spectrum distinct or do they overlap?

The sectors (0 to 30°), (30° to 90°), (-30° to 0°) and (-90° to -30°) will overlap (by approximating $0.52/\lambda$ to $0.5/\lambda$ otherwise instead of 30° it would be 31.4°). These angles are with respect to the array axis. Therefore, they correspond to 8 sectors in the azimuth plane. One way that this ambiguity can be resolved is by having the transmitter antenna rotational information available also.

The vhf antennas will have to have RF amplifiers mounted on them to boost the signal level. Such an antenna amplifier has been described in [21]. After mixing with a stable local oscillator and IF amplification stages, whose gains are adjusted to obtain necessary taper, coherent detection is performed by multiplying the signal with the inphase and quadrature outputs of an oscillator phase locked to the transmitter pulse. After lowpass filtering we obtain the inphase and quadrature components of the signal received.

We have now obtained 32 spatial samples. Each sample consisting of a real and imaginary part. These must now the be digitized. Since the transmitter pulse width is 6 μs the range gate will also be of 6 μs duration. Thus these 32 samples must be processed every 6 μs. If we want a slack of 1 μs then

the conversion must be done in 5 μs. The AD 574 A/D converter has a conversion time of approx. 30 μs for 8 bits. Thus 6 A/D converters each for real and imaginary parts would be required at each of the 32 receiver outputs giving a total requirement of 384 A/D convertors. These 6 A/D convertors would each deal with every sixth range cell. The receiver output being connected to each in a round-robin fashion (see pg 57). The output of the A-D convertor would have to be switched similarly to maintain the range cell sequence.

We now have 32 samples each of 16 bits (real and imaginary) these will be updated every 5 μs. A DFT is performed on these samples. The algorithm used is the one given in Fig. 11 (pg 59) of [22]. Ease in coding a simulator for samples which are a power of 2 in number was the reason for the choice. For each of the 5 stages we could have 32x16 (or 12) bit locations for storing the results of each stage. In which case our requirement is that each stage be finished in 5 μs. In case we use only one set of 32 locations then the complete DFT has to be over in 5 μs.

In this implementation it is proposed not to use any hardware multiplier. Instead a table look-up with a adder and hardwired shift is proposed to be used. For each weight, 16 in this case, we have a table with 16 entries. Each entry corresponds to the product of ten weight with the integers 0 through 15. The integers 0 through 15 represent the numbers that can be represented by 4 bits. If we have a 12 bit

number in whose product with the weight we are interested, instead of doing 11 shifts and adds (assuming that the weight does not have less than 12 bits) we take the first 4 bits and use it as address for the table look-up. The table contents are read out into one of the inputs of the adder. The eight most significant bits of the output goes to a latch which is connected back to the other input. Now the next 4-bits are read and similarly acted upon. After last four bits are acted upon, the adder output goes to an output register (taking care of the sign bit) the latch is cleared. Now we require just 3 adds. By making 3 copies of a table and dedicating each to a 4-bit portion the address look-up can be simultaneous. The output being held in registers/latches which are switched to the input of the adder in sequence.

The flow-chart for the implementation of this DFT as part of its simulation is shown in Appendix A.

A word about the DFT simulator. Its purpose is to determine what effects different register size, memory (table) word size and quantization levels will have on the result as compared to the results from a DFT using a software FFT based on the same algorithm. Towards this end the parameters DATPOS and ~~MEMPOS~~ POSMEM (~~MEMPOS~~ POSMEM has to position the memory word in the most significant bits of the register) are used so that a smaller memory or dataword may be appropriately positioned

in the register word to get the best result. Fixed point arithmetic is used. It was found that by scaling the data word such that maximum data word was less than maximum register content by a factor of 16. i.e. DATPOS positioned the data word in the register after leaving the most significant four bits, no overflows resulted with no requirement of scaling in the middle of the processing.

We now consider the timing for addressing the table and operation of the adders which will determine the memory requirement and the number of adders.

The schematic for the multiplier implementation is shown in Fig. 8.

From the spec. sheets of DM74S271 (256x8 bit ROM), SN 74/S283 (4 bit adder) and SN 74S412 (multimode buffer latch) taking maximum times, we see that 60 ns are required for the table look-up then 27 ns for the latch output then 22.5 ns for 12 bit add again 27 ns in the latch and then 22.5 ns for adding. Then the output from the output latch is taken after 27 ns. This total comes to 189 ns. further 13 bit addition we take 25 ns. This makes the total 214 ns. If we take 30 ns for each complementing we get a total of 274 ns.

Now of the 16 weights two, namely for 0° and 90°, can be hardwired, since for 0° the weight is 1 and for 90° the real

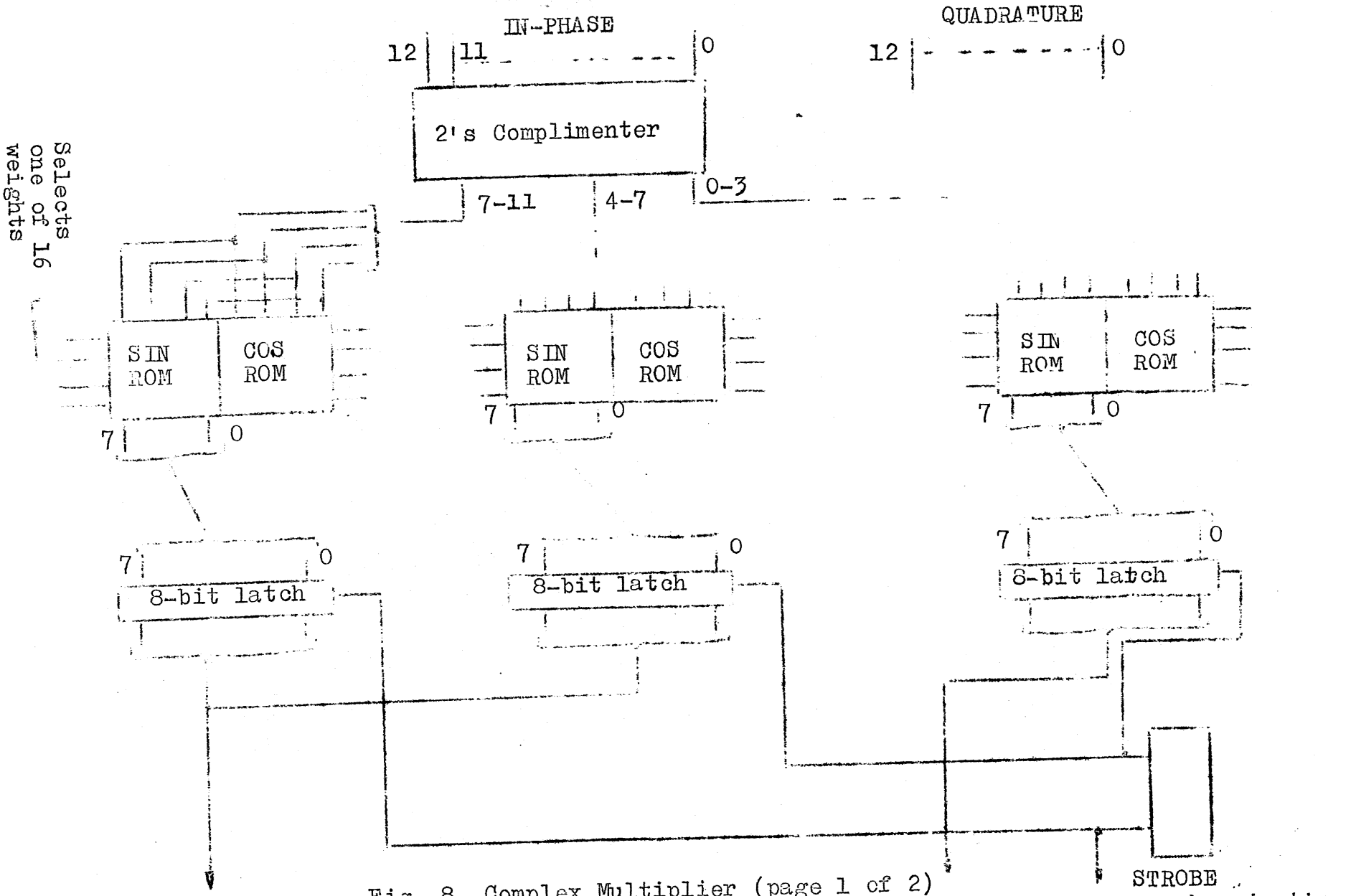


Fig. 8 Complex Multiplier (page 1 of 2)

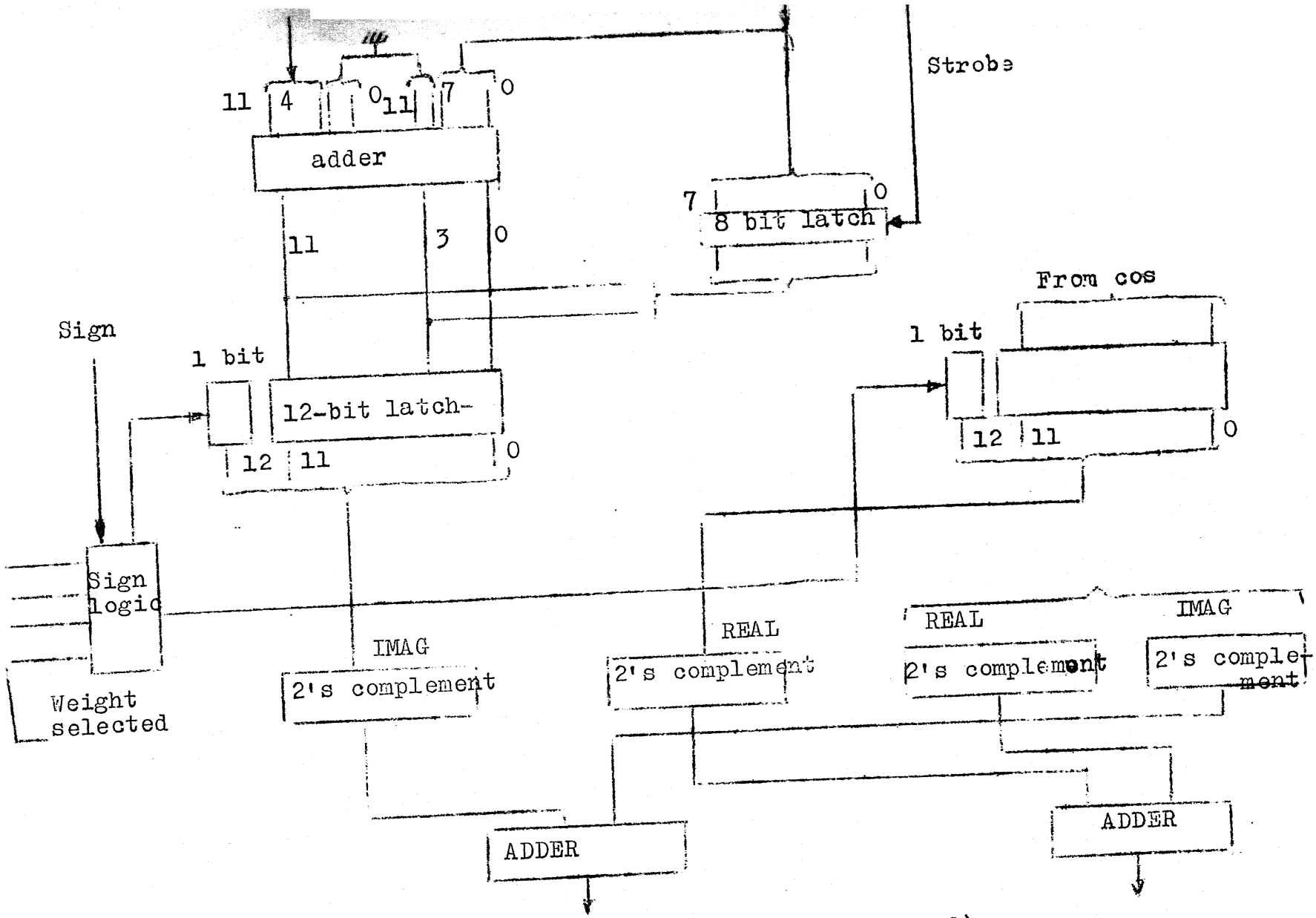


Fig. 8 Complex Multiplier (page 2 of 2)

and imaginary parts are switched (after changing the sign of the real part). Thus we have 14 weights which have to be multiplied and 5 μs to do it in, that is for each weight we have 5 μs/14 ≃ 330 ns. We therefore have a leeway of (330 - 274 ns ≃ 55 ns). Thus it would seem possible to multiplex the same for all the 14 operation to complete the first stage of the DFT. The second stage will require 12 such multiplications. The third stage would require 8. The fourth and fifth stages are hardwired and would require only as much time as required for doing the butterflies.

Thus for the first three stages we require three multipliers and since the third multiplier is required only for 8 operations. It may share the load of the 1st stage multiplier. Thus we will have two multipliers doing 11 weighting operations and one multiplier doing 12. Thus for each complex weighting we have approximately 400 ns (≃ 5μs/12). Thus we have a leeway of 400 - 275 = 125 ns, for each weighting operation, to take care of the multiplexing and timing.

A list of the ~~chips~~ hardware required for the above implementation of the DFT is given at Appendix B.

4 THE MTI Filter

A moving target will produce a doppler frequency which because of the sampling at the p.r.f. rate will cause an aliasing about $\frac{1}{2}$ p.r.f. The blind speed for the system considered is

$$V_R \cdot \frac{1}{\text{p.r.f.}} = n\lambda/2 \qquad \text{where } \lambda = 2 \text{ m., p.r.f.} = 400 \text{ pps}$$

$$V_R = \frac{2}{2} \times 400 = 400 \text{ m/s} = 1440 \text{ km/hr.}$$

It is seen that the first blind speed is very high caused by the use of a v.h.f. frequency. This is an advantage.

To detect a moving target and suppress the response from stationary/low doppler velocity targets the consecutive outputs obtained at each of the 32 angular positions, is passed through a digital filter.

The flow chart for the digital filter is shown in Fig. 89 It is based on a 3 stage cascade of 2nd order sections in direct form 2[as mentioned on page 43, Ref. 22]. To determine the values of the various coefficients the following method was adopted. A unit sequence (in time domain) i.e. 1 followed by 255 zeros. was given as input to the digital filter routine. The output sequence from the digital filter routine was fed to a FFT subroutine. The magnitude of the 256 point frequency domain output represents the filter transfer function at each of the 256 points between 0 and p.r.f. The magnitude is fed

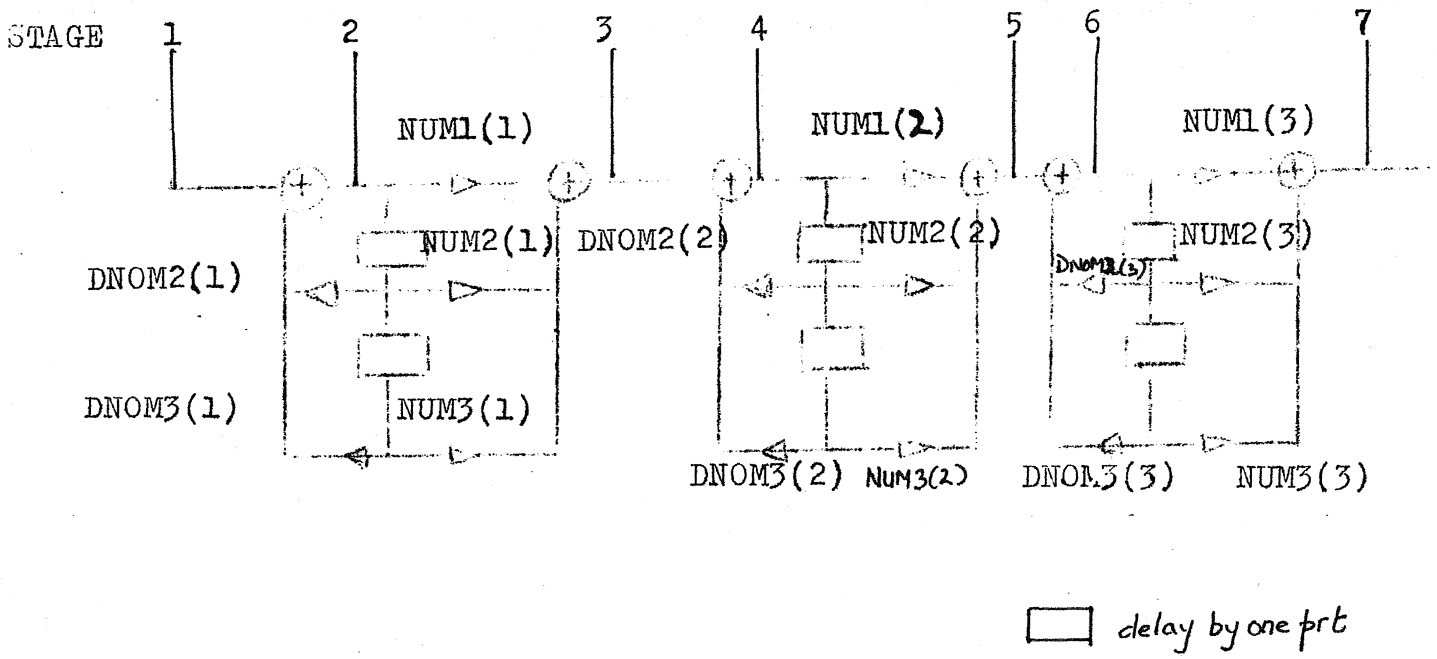

STAGE
1
2
3
4
5
6
7
NUM1(1)
NUM1(2)
NUM1(3)
NUM2(1)
DNOM2(2)
NUM2(2)
NUM2(3)
DNOM2(1)
DNOM3(1)
NUM3(1)
DNOM3(2)
NUM3(2)
DNOM3(3)
NUM3(3)
delay by one prt

FIG 9(a)

Filter Flow Chart

Step 1 : Read in values of coefficients NUM1, NUM2, NUM3, DNOM1, DNOM2, DNOM3.

Step 2 : Read in input sequence into stage 1.

Comment : The routine uses enough memory so that at each stage the complete sequence is retained instead of the previous two values only. It could just as easily have been done with two storage location. But for purposes of debugging and the initial conceptualization for coding the complete sequence was operated and shifted down. Thus INTER (7,256) stores the values at the 7 stages shown.

Step 3 : Process the first pulse through the 7 stages.

Step 4 : Process the second pulse through the 7 stages. The delay units have now been filled.

Step 5 : Send the subsequent pulses through the 7 stages one by one.

Step 6 : Delete or weight the first 3/4 outputs.

Step 7 : Return.

FIG 9(b)

to a plot routine which displays the log of magnitude versus the discrete frequency on the CRT terminal of the computer.

The method is analogous to tuning a hardware circuit using a signal/pulse generator and an oscilloscope. Needless to say it requires a little patience. The coefficients of the numerator decide the cut off region by appropriately weighting the denominator a pole can be positioned so that the transition region slope can be made steep. The frequency response of the filter and the coefficient used are given in Appendix G.

If the MTI was required over a range of say 100 kms involving approximately 100 range cells each range cell would require 6 storage locations for the digital filter stage plus 1 for the input. The digital filter would have one PRT ($\frac{1}{PRF} \approx$ 2500 μs) to attend to all the 100 range cells, i.e. approximately 25 μs per range cell. There can be 32 such filters to attend to the 32 angular positions. The MTI storage location required would be 32x7x100 = 22,400 storage locations for complex numbers. As the DFT unit puts out a 26 bit complex number at each angle position the total storage requirement would be 22,400x26 bits.

By using a look-up multiplier as outlined for the DFT or a hardwired multiplier it should be possible to do each 26 (complex) x 8/9 bit multiplication in $<$ 500 ns.-for the 8 complex multiplications this means 6 μs. As each cell has

got 25 μs to be processed there is sufficient time to complete the 12 complex additions.

The MTI implementation would be the same as a digital MTI in a conventional radar only in this case 32 of them would be required to operate on the 32 angle outputs.

The output from the last adder is in complex form, this is to be converted into a magnitude. There is 25 μs to do this using two 16x16 bit multipliers and one adder it should be possible to do the two multiplications (squaring) and additions in 1-2 μs.

After checking against a threshold this information is fed to the 8 sectors which are aliased. A logic circuit which determines the sector illuminated by the transmitter based upon its synchro information will enable the corresponding sector and disable the others. The details of extraction of angle information and the logic circuit are not discussed.

An outline sketch of an MTI filter has been given. The design has not been simulated, though the algorithm has. It is felt that ground returns at ranges in excess of 30 kms will not be excessive and could be taken care of by using STC (sensitivity time control) and biasing using a clutter map.

CHAPTER 4

SIMULATION

The program for the simulation is given at Appendix C. The algorithm is given below.

Step 1 : Input antenna parameters. Input DFT simulation parameters, Target and clutter parameters, Switches LIMTR2, LIMTR, TAPER, CONV.

Step 2 : Load the ROMs for passing to DFT routine.

Step 3 : Form the angular information to be passed to the PLOT sub-routine.

Step 4 : Form the parameter to be passed to the RXROUT sub-routine which simulates the limiting action of the IF amplifier.

LOOP :

Step 5 : Call noise subroutine (returns value for each of the 32 antennas). Add doppler phase shift.

Step 6 : Vector addition of clutter target signals and noise.

Step 7 : If LIMTR call RXROUT.

Step 8 : If LIMTR2 call RXR1.

Step 9 : If TAPER then begin if antenna unserviceable WT = 0 else $WT = 0.54 + 0.46 \cos\left(\frac{\pi \cdot ANT}{32}\right)$ end. TAPER the 32 samples. (ANT gives the antenna position-- it will take values -16 to +16).

Step 10 : Store in array.

Step 11 : If all antennas not done go to Step 6.

Step 12 : If CONV/ then begin call A2D routine, call FFTSIM end.

Else call FT routine.

Step 13 : Determine the amplitudes and phase of the spectrum return by the DFT routine.

Step 14 : Store these values.

Step 15 : All return processed? If not, go to LOOP

Step 16 : Plot amplitude.

Step 17 : Plot phase.

Step 18 : Call DIGFIL.

Step 19 : Determine magnitude. Average the magnitudes over the complete sequence.

Step 20 : Plot magnitude against angle.

Step 21 : STOP.

It is assumed that the extreme left antenna is the phase reference point and that targets and clutter are all located at equi delay locus from this reference point. This would not in any way effect the results of the simulation.

We can simulate the effects of noise level, local oscillator instability, inaccuracy in antenna position as also the DFT parameters.

The simulation is only for one set of targets in a given range cell as the simulation is for verifying the angular resolution system proposed.

.1 The Noise Subroutine (AppendixDD)

White Gaussian noise samples should have a constant magnitude in discrete frequency spectra. Therefore we take such a spectra. The phase of the spectral components we take to be uniformly distributed from 0 to 2π. This complex spectrum we now inverse transform to obtain the time series. This is now our noise sequence. That the real and imaginary components will be Gaussian follows from the central limit theorem, the number of spectral components taken being 32.

4.2 RXR1 Subroutine :

This subroutine simulates the limiting action of the IF amplifier and the coherent detection by multiplying it with cos (wt) and - sin (wt). The 1000 points are taken over a cycle multiplied and summed to determine the output of the coherent detector. The symmetries in the sin and cos functions are exploited as can be seen from the routine given in Appendix E.

4.3 RXROUT Subroutine : (Appendix F) :

Utilizes a 4th order Runga Kutta integration to do the same job as RXR1. It was used to check the performance of RXR1. The time taken by this is about 1.52 secs whereas RXR1

takes about 0.1 secs.

Signal to Noise Ratio :

We must determine what must be the spectral amplitude of the discrete noise samples that must be added to the signal. These discrete noise sample are taken at the PRF rate that being the rate at which each range cell is sampled.

Considering a target of $1m^2$ at a range of 150 km. The transmitted peak power is 750 kw. If the transmitter antenna beamwidth in the horizontal plane is 15° and the vertical 60°. Then, as

$$G_D \simeq \frac{41000}{\theta_B \cdot \varphi_B} \quad \text{[Ref. Skolnik Radar Handbook 1, pg 9-5]}$$

$$G_D = \frac{41000}{15 \times 60} = 45.5 \simeq 16.6 \text{ db}$$

The signal peak power returned from the target

$$= \frac{750\times10^3 \times 45.5 \times 1}{(4\pi)^2 \times (1.5)^4 \times (10^5)^4} = \frac{34000\times10^3\times10^{-20}}{5.1 \times 154} \approx 43.3\times10^{-17} \text{ watts/m}^2$$

If the receiver dipole has a gain of 0 db and an efficiency of 0.5 its effective capture area $= \frac{\lambda^2 G \cdot \eta}{4\pi}$; $\lambda = 2m$

$$= \frac{4}{4\pi} \times 1 \times 0.5 = 0.159 \text{ m}^2$$

Therefore the signal received at one antenna $= 0.159\times43.3\times10^{-17}$ wa

$$= 6.8\times10^{-17} \text{ watts}$$

We have the relation between output signal to noise ratio and the input signal to noise ratio as

$\frac{(S/N)_i}{(S/N)_o} = F_n$ where F_n is the noise figure of the receiver by definition.

$(S/N)_o = \frac{S_{min}}{kT_oBF_n}$ where B is the bandwidth of the receiver

k - Boltzmann constant

$kT_o = 4x10^{-21}$ watts/Hz assuming $T_o = 290^oK$

Assuming F_n = 9 db and since pulse width is

$$\tau = 5\ \mu s \quad B \approx \frac{2}{\tau} = 400\ kcs.$$

We have

$$(S/N)_o = \frac{6.8x10^{-17}}{4x10^{-21}x4x10^5x8}$$

$$= .053x10^{-1} = .005$$

Therefore, N/s = 200

N = 200 x S

Since the manner in which we generate the noise leads to 32 spectral lines in the range 0 - 400 c/s (prf), each of these spectral lines will have a power of 200xS/32 = 6.25 x S, in other words their amplitudes will be $\sqrt{6.25}$xampl. of signal = 2.5 x Sig. amp.

Thus the spectral amplitude/signal amplitude ratio must be $\geq$ 2.5 when they are added prior to simulating the limit IF amplifier and coherent detection.

If only 16 pulses were processed then ratio $\geq 2.5 \times \sqrt{2} \simeq 2.85$. If 8 pulses then it should be $\geq 2.5 \times 2 = 5$. However higher ratios are considered to partially offset the fact that target scintillation has not been taken into consideration.

.5 Clutter to Signal Ratio :

As the transmitted beam width is 15° ($\simeq$ 1/4 radian) the patch of ground at range R illuminated during a pulse width, assuming the complete beam hits the ground, is

$$R\tau C/4x2 = \frac{Rx6x10^{-6}x300x10^{3}}{8}$$

$$= 0.225xR \text{ sq.km.}$$

at 100 kms the patch would be 22.5 sq. km. Taking the scattering cross-section as -30 db [24], the clutter cross-section would therefore be $22.5x10^{3}$ sq.m. Thus compared to a $1m^2$ target at the same range the total clutter power would be $22.5x10^{3}$ times greater or the total clutter voltage would be 150 times greater. The simulation has provision for 5 point-clutter whose azimuth separation from the target, relative strength and doppler can be read in. Thus each clutter point

will have to have a voltage approximately $\sqrt{\frac{22.5}{5} \times 10^3} \approx 67.5$ tir times that of the signal voltage. Admittedly this is much too simplistic a clutter model. However, it is unlikely that at the ranges under consideration the complete beamwidth would illuminate a continuous patch of clutter. The above however, gives an idea of the clutter signals to be used.

4.6 Number of returns from a Target

Since the transmitter antenna rotates at 6 rpm and has a beamwidth of 15° the number of returns from a point target for 400 transmitted pulse/sec.

$$= \frac{60 \times 15 \times 400}{6 \times 360} = \frac{1000}{6} = 166 \text{ pulses}$$

The simulation processes 8/16 returns this is to facilitate the assumption that over a small section of the beam the beam-shape and scanning effects can be neglected.

CHAPTER 5

CONCLUSION AND SUGGESTIONS

5.1 CONCLUSIONS

The simulation has brought out

a) The requirement of preventing the signal amplification stages from limiting.

b) That under expected noise levels and clutter levels (30:1) it is possible to determine the azimuth angle.

c) That the full conversion range of the A-D converter should be utilized by incorporating appropriate biasing.

d) That register word lengths of 12 bits (11+1 sign), and 8 bits in memory tables should give adequate performance.

e) Owing to endfire resolution being poor two linear arrays each of 32 elements and mutually perpendicular would be required.

In conclusion let us list the major system features as envisaged.

a) Two mutually perpendicular antenna systems so as to avoid loss of resolution along the endfire direction.

b) Synchro information from the transmitting site to be transmitted to the receiving site for deciding the display sector. The other information is trigger and a lock pulse for the coherent oscillator.

c) RF amplifier to be antenna mounted.

d) An IAGC circuit and/or a biasing voltage proportional to the clutter to prevent the IF amplifier from limiting.

e) Coherent detection followed by DFT unit and possibly a MTI unit followed by display unit controlled by logic circuit for determining the sector. The max delay caused by the DFT and MTI processing is expected to be 50 μs.

f) A switching arrangement which will enable the mixer and subsequent stages to be shared between the two mutually perpendicular arrays.

An overall system block schematic is shown in Fig.10 (pg57)

An apparent disadvantage may be that for given transmitted power the return from a target will be less for the antenna with a wider beam (lower gain) while the ground clutter assuming it to be uniform would return the same power. Thus the input signal/clutter ratio would be poor and hence sub-clutter visibility at the output of an MTI system would also be poor. But what has been proposed is for a system where the beam has been made as narrow as operationally feasible and one is now trying to look inside the beam. The input signal/clutter

ratio thus will in any case be the best possible under the circumstance.

The scheme though proposed for a vhf system could in principle be applied to system at higher frequency. For example for a radar in the L band (λ = 0.3 m) with a beam width of 2° would require sampling at approximately $.07/\lambda$ samples/metre If sampling frequency were divided into 8 discrete points the 8 spatial samples would be ~~spa~~sprcad out over $8 \times \frac{\lambda}{.07} = 240/7 \simeq 34$ metres. At 60° from broad side such a system would still give a beam width of approximately 1°. At broad side the beam width would be approximately 0.5°. Of course the number of sectors which have to be resolved has increased.

5.2 Suggestions

Since in any array there will be some interelement coupling one convenient way of compensating for this would be by multiplying each of the samples by a compensating factor before it is fed to the DFT. unit.

By taking the concept of spatial frequency and the band limiting action of the transmitter antenna, we have by applying conventional signal processing, namely a DFT, been able to obtain angle information from within the beam. It may be interesting to consider other signal processing methods (e.g. an elliptic filter) which will give narrower passbands

and could be used in a given direction.

By introducing a threshold check before A-D conversion it may be possible to get more time for processing by rejecting those sample sets where a given number of samples fall below the threshold.

It may be worthwhile to implement the DFT as well as the MTI filter using CCD devices. This would at least cut down on the A/D conversion. Electro-optical array processors have also been reported [23].

An important aspect that will have to be considered in any final design is the monitoring of various parameters and the incorporation of automatic self-check features.

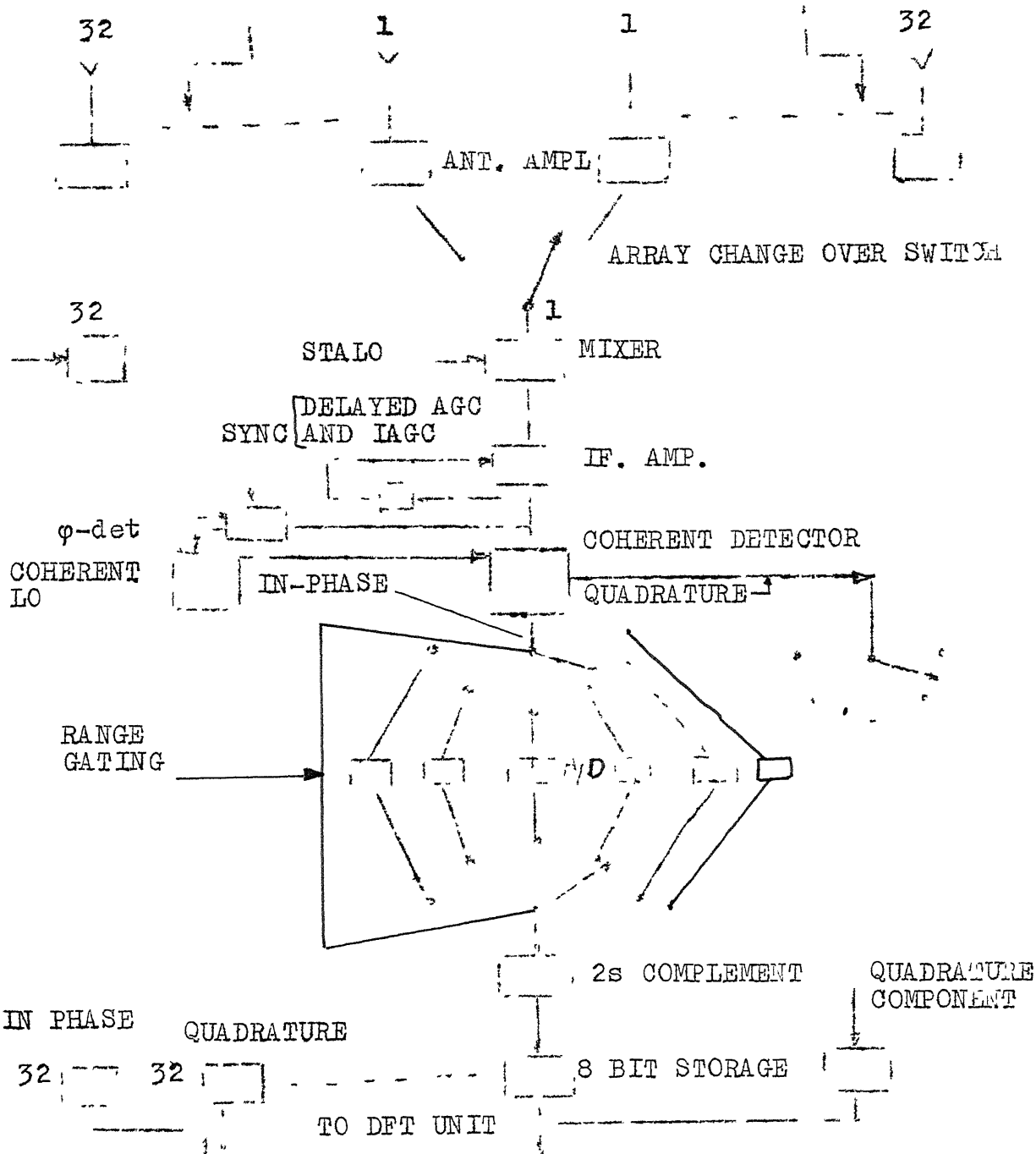
Evaluation of probability of detection and probability of false alarm has not been done. The simulator that has been coded may be extended to determine these for different values of SNR, under conditions of scintillating target and a more appropriate clutter model.

5.3 Discussion

How does this method compare with the one proposed by DEN DAVIES et al [25]? The method proposed by Davies et al uses a swept frequency local oscillator to introduce the phase, the mixer outputs are then compared in an IF beam forming network to obtain the received beam. As the transmitted beam

rotates the local oscillator phase shift would have to be adjusted so that the received beam sweeps within the transmitted beam. Secondly assuming that 32 received beams are formed in both cases. In the case where beam forming is done by sweeping the local oscillator only 1/32 of the total energy (over one pulse width) received from a target is utilized - assuming that the received beam looks in each of the 32 position for the same length of time. Thus the video output after detection and lowpass filtering will be much lower. Whereas in the proposed system as the total received energy is stored and then appropriately combined using the DFT no such loss as caused by 'distructive' combination can be expected. Thus to get the same performance of detection the power transmitted in the case of the former method would have to be 32 times greater or the max. range reduced to $1/\sqrt{32} \simeq 0.42$. Using the method proposed of course would involve coherent detection for each of the 32 antenna outputs which would not be the case with the method proposed by Davies et al.

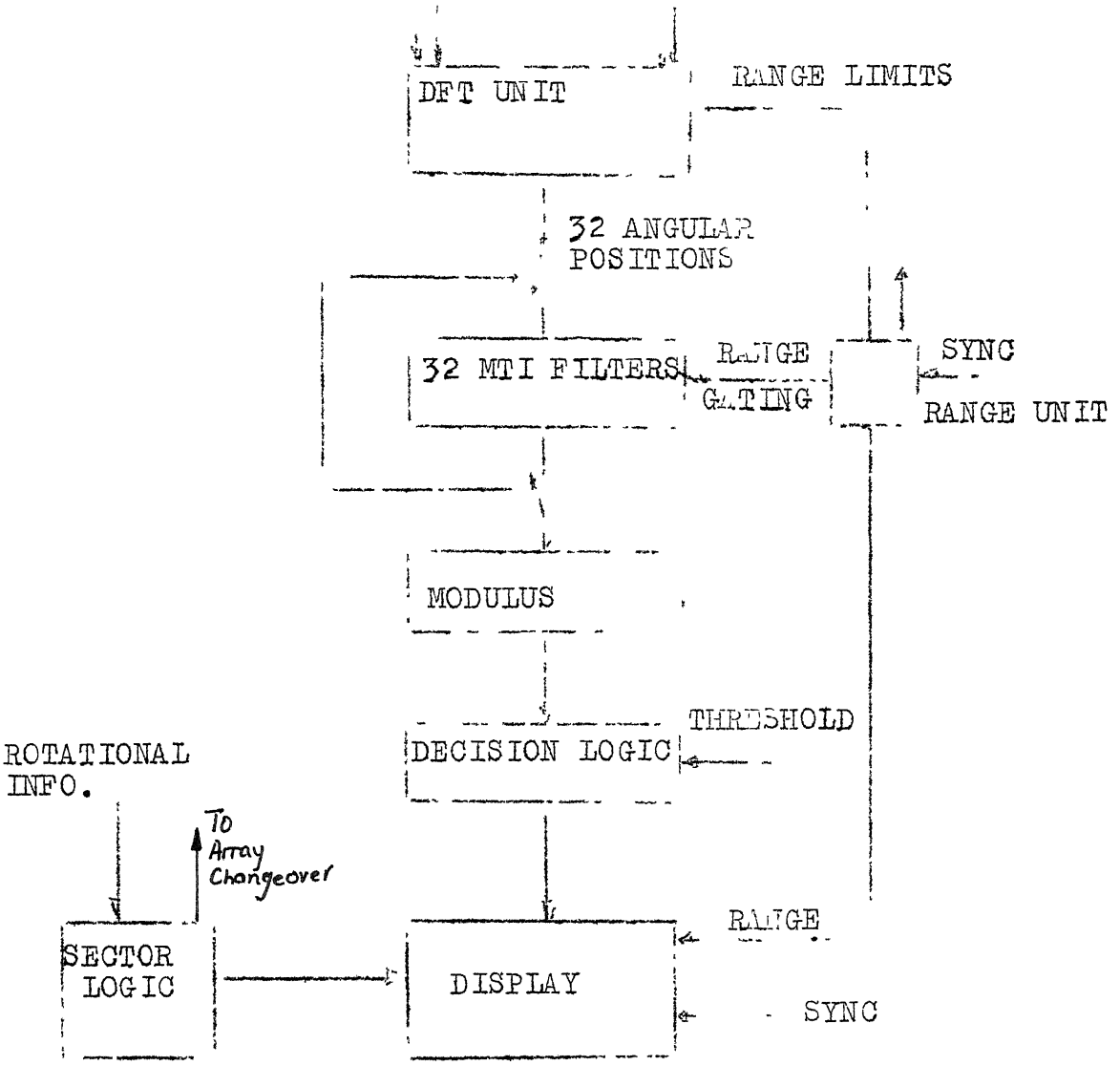
[27] proposes a holographic radar suitable for high range resolution at small ranges, the application being measurement of ice thickness. The radar operates in CW mode and uses the holographic effect for determining the range (depth). The present proposed processing resembles it in that the output after coherent detection is converted and stored and subsequently the beam is formed.



Page 1 of 2

OVERALL SYSTEM BLOCK SCHEMATIC

FIG 10



Page 2 of II

OVERALL BLOCK SCHEMATIC

Fig 10

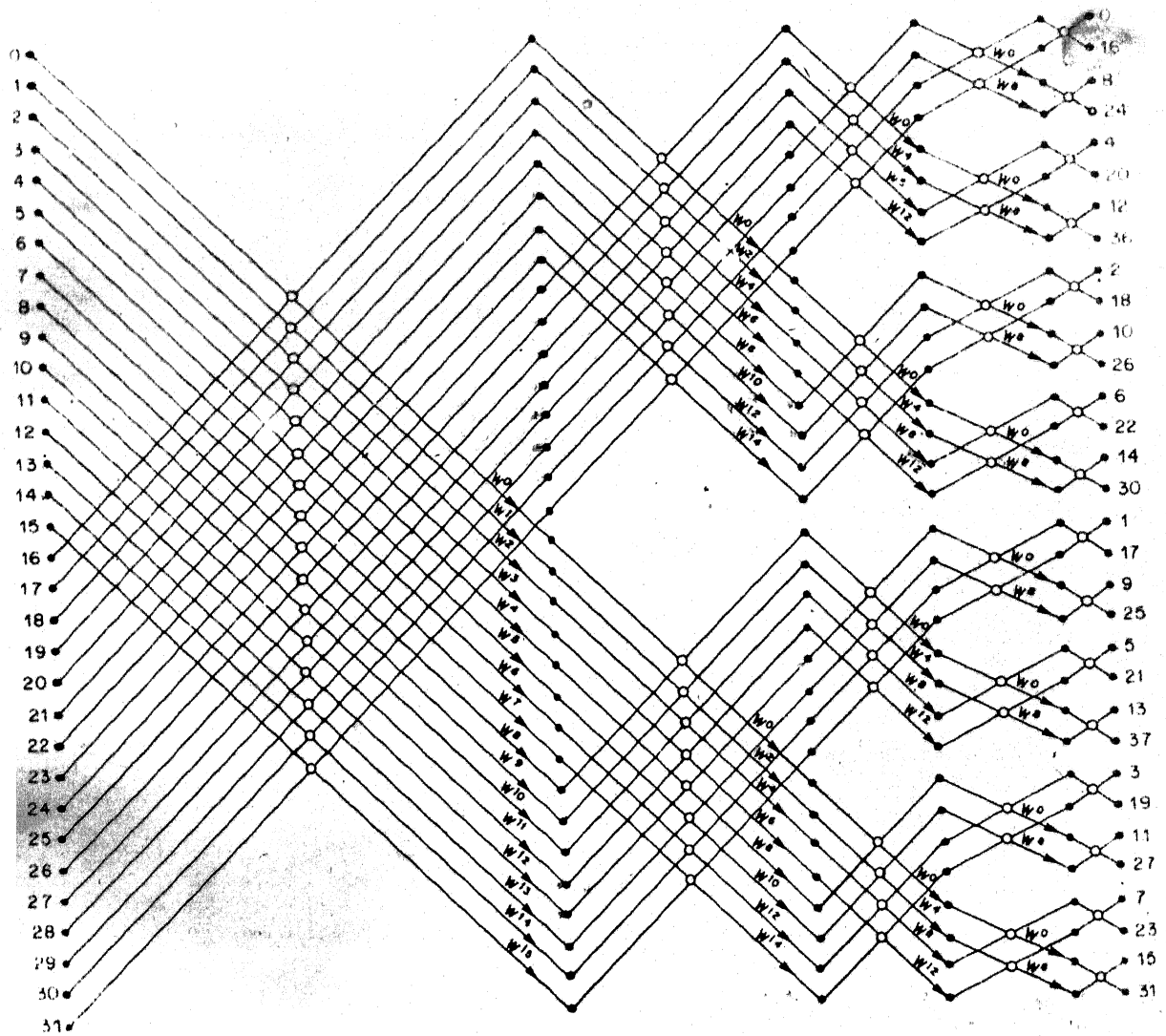


Fig. 6.14 32-Point, radix 2, in-place FFT.

FIG 11. (from [22])

REFERENCES

1. C.E. MUEHE et al. 'New techniques applied to air traffic control radars', Proc. IEEE, Vol. 62, No.6, pp 716-723, June 74.
2. E.N. GILBERT and S.P. MORGAN, 'Optimum design of directive antenna arrays, Subject to Random Variations', Bell Sys. Tech. J. Vol. 34, pp. 637 663, May 1955.
3. S.A. SCHELKUNOFF, 'A mathematical theory of linear arrays', Bell Sys. Tech. J. Vol. 22, pp. 80-107, Jan. 1943.
4. E.H. NEWMAN et al., 'Superdirective receiving arrays', IEEE Trans. Ant. and Prop. Vol. AP-26, No.5, pp 629-636, Sep. 78.
5. M.I. SKOLNIK, 'Resolution of angular ambiguities in radar antennas with widely spaced elements and grating lobes', IRE Trans. Ant. and Prop. Vol. AP-10, pp 351-352, May 62.
6. W.E. KOCK, 'Related experiments with soundwaves and electromagnetic wave', Proc. IRE, Vol. 47, pp 1192-1201, Jul. 59.
7. B.S. McCARTNEY, 'Theoretical and experimental properties of two element multiplicative multi-frequency receiving arrays including superdirectivity', The Radio and Electronic Engineer, Vol. 28, No.2, pp 129-143, Aug. 64.
8. N. FARHAT, 'A new imaging principle', Proc. IEEE, Vol. 64, pp 379-380, Mar. 76.
9. D.S. BAGRI, 'High resolution interferometric observation of radio sources at 327 MHz', Ph.D. Thesis, Univ. of Bombay, Nov. 1975.

10. H.F. HARMUTH, 'Synthetic aperture radar based on non-sinusoidal function-I', IEEE Trans. of Electromagnetic Compatibility, EMC-20, pp 426-435, Aug. 78.

11. H.F. HARMUTH, 'Synthetic aperture radar based on non-sinusoidal function-II', IEEE Trans. of Electromagnetic Compatibility, EMC-21, pp 40-49, Feb. 79.

12. G.W. STROKE, 'Optical computing', IEEE Spectrum, Vol.9, No.12, pp 24-41, Dec. 72.

13. W.E. KOCK, 'Holography helps radar find new performance horizons', Electronics, Vol. 43, pp 80-88, Oct. 12, 1970.

14. W.E. KOCK, 'Sidelooking radar holography and doppler-free coherent radar', Proc. IEEE, Vol. 56, pp 238-239, Feb. 68.

15. W.E. KOCK, 'Stationary coherent (Hologram) radar and sonar', Proc. IEEE, Vol. 56, pp 2180-2181, Dec. 68.

16. J.W. GOODMAN, 'Introduction to fourier optics', McGraw-Hill, 1968.

17. C.F. AUGUSTINE et al., 'Microwave holography using liquid crystal area detector', Proc. IEEE, Vol. 57, pp 1333-1334, July 69.

18. SKATTEBOL, et al., 'Electronic reconstruction of acoustic hologram', IEEE Trans. Aud. and Elect. Acoustics, Vol.AU-18, No.1, pp 59-62, Mar 70.

19. P.W. HOWELLS, 'Exploration in fixed and adaptive resolution at GE and SURC', IEEE Trans. Ant. and Prop. Vol. AP-24, No. 5, pp 575-564, Sep. 76.

20. D.C. COOPER, 'The use of an effective transmission pattern to improve the angular resolution of within the pulse scanning system', The Radio and Electronic Engineer, Vol. 28, No.3, pp 153-160, Sep. 64.

21. J.R. COPELAND et al., 'Antennaffier arrays', IEEE Trans. Ant. and Prop. Vol. AP-12, pp 227-233, Mar. 64.

22. LAWRENCE R. RABINDER and BERWARD GOLD, 'Theory and application of digital signal processing', Prentice Hall India,1978.

23. T.W. COLE, 'Electro-optical array processor for complex signals', App. Opt. Vol. 17, No.18, pp 2952-2955, 15 Sep. 78.

24. RUCK, BARRIC, STUART, KRICHBAUM, 'Radar cross-section handbook', Vol. 2, Plenum Press, 1970, pp 768, Table 9.2.

25. D.E.N. DAVIES et al., 'A high resolution radar incorporating a mechanical scanning transmitter and a static multibeam receiver', IEE International Conference - RADAR 77, Conference publication No. 155, pp 59-62.

26. M.I. SKOLNIK, 'Radar handbook', McGraw-Hill, 1970.

27. K. IIZUKA et al., 'A hologram matrix radar', Proc. IEEE Vol. 64, No. 10, pp 1493-1504, Oct. 76.

28. N. FARHAT, 'High resolution microwave holography and the imaging of remote moving objects', Optical Engineering, Vol. 14, No.5, pp 499-505, Sept./Oct., 75.

LIST OF APPENDICES

Comment : The appended listings of programs show a number of statement as a comment. These were introduced for debugging. All subroutines have a test program which was used for program development and debugging.

APPENDIX A4

FLOW CHART OF DFT SIMULATOR

Step 1 Input the following parameters : 1) No. of aerials/ spatial samples (IORD), 2) NSTAG log of spatial samples in base 2, 3) No. of quantization levels (KWANT), 4) WRDSIZ wordsize corresponds to the number of bits in the registers which will hold the output of each stage as also the adders, 5) No.of bits for each entry in the table MEMWRD, 6) POSMEM appropriately positions the bits of the table look up in the registers, 7) DATPOS positions the data bits in the register, 8) DCDBIT the number of bits used for address decoding.

Step 2 Load the tables with 2 * * DCDBIT entries for each of IORD/2 complex weights for purpose of simulation rotate the bits corresponding to MEMPOS.

Note : When calling from main routine Step 1 and 2 are passed as parameter Step 3 is not used.

Step 3 Set random number generator generate (IORD) real and imaginary samples and quantize them put them in arrays STOR and STOI. Retain the unquantized numbers in array XTEST for verification. For purposes of simulation the contents of STOR and STOI are rotated to correspond to DATPOS.

Step 4 Commence outer loop of FFT simulator. ISTAG taking values 1 to NSTAG. N1 = ISTAG -1; N2 = 2 ** N1; N = 2 ** (NSTAG -1)/N2. N2 is now the number of loops in each stage of the NSTAGs stages. N is the number of butterflies in each of the N2 sections.

Step 5 Do the loops corresponding to each stage. No. of loops for I1 taking values I1 to N2. N3 = (I1 - 1) * N * 2. N3 will give the address for register for the first input to the first butterfly in each of the N2 loops.

Step 6 The inner most loop. For I2 taking values 1 to N - N is the number of butterflies as given above. N4 = N3 + I2, N5 = N4 + N, IN = 2 ** (ISTAG - 1) * (I2 - 1) + 1. IN is used in determining the appropriate weight in Step

Step 7 Real and imaginary butterfly operation. Real

BTFLYR(I2,1) = STOR(N4) + STOR(N5)

BTFLYR(I2,2) = STOR(N4) - STOR(N5)

Imaginary

BTFLYI(I2,1) = STOI(N4) + STOI(N5)

BTFLYI(I2,2) = STOI(N4) - STOI(N5)

Step 8 Obtain the sign of BTFLYR(I2,2) and BTFLYI(I2,2) as these will be involved in multiplication by the weights.

Step 9 Check for error. If any of the butterfly outputs will result in register overflow then go to ERROR.

Step 10 Put the sum output of the butterflies back in STOR(N4) and STOI(N4) respectively.

Step 11 If ISTAG equal to NSTAG then store the difference outputs also as no weighting required in last stage and go to Step 24.

Comment : We have 4 adders two each for real and imaginary parts. They each have two inputs represented as ADIR11, ADIR21 for No. 1 real adder. ADIR12, ADIR22 for No.2 real adder. ADII11, ADII21 for No.1 imaginary adder. ADII12, ADII22 for No.2 imaginary adder. The corresponding outputs are ADDR1, ADOR2 and ADOI1, ADOI2. No.1 real and imaginary adders handle the product of the real part with the complex weight. No.2 real and imaginary adders handle the product of the imaginary part with the complex weight.

Step 12 Clear the inputs to the adders.

LOOP

Step 13 Get the output of the adders.

Step 14 Shift each output to the right by the number of decode bits DCDBIT.

Step 15 Extract the least significant decode bits (which have as yet not been used) from the difference output of the butterflies, i.e. BTFLYR (I2, 2) and BTFLYI(I2,2).

Step 16 Use these bits as addresses for looking up the table corresponding to the appropriate weight.

Step 17 The result of the table look up is put into input 1 of the appropriate adder. ADIR11 gets the product of the real part with the real part of the complex weight. ADII11 gets the product of the real part with the imaginary part of the complex wiweight. Similarly, ADIR12 gets the product of the imaginary part with the real part of the complex weight. ADII12 gets the product of the imaginary part with the imaginary part of the complex weight.

Step 18 If the most significant decode bits have not been used go to step 13.

Step 19 Form the adder outputs and affix the appropriate signs saved in Step 8.

Step 20 Form the real and imaginary part of the product by appropriately adding and subtracting the four products obtained as output of the adders in Step 19.

Step 21 Check for overflows. If ifany overflow go to ERROR.

Step 22 Put the result in appropriate storage, i.e. STOR(N5) and STOI(N5).

Step 23 END of inner loop. End of middle loop. End of outer loop.

Step 24 Do bit reversal and shuffle the storage accordingly.

Step 25 Scale output down by shifting by DATPOS bits.

Step 26 Form the magnitude of the complex output.

Step 27 Call FFT routine using input as XTEST for verification.

Step 28 Check error of corresponding samples.

Step 29 Stop.

APPENDIX B

HARDWARE REQUIRED FOR MULTIPLIER

Item	Qty
ROM DM74S271	12
8 bit latches SN74S412	6
12 bit latches (two SN74S412)	12
12 bit adder (three SN741S283 4 bit adders)	6
13 bit 2's complementer	12
1 bit latch	2
SIGN logic circuit	
12x1 multiplexer	26
Output demultiplexer	26
Butterflies	
Adders	5
Subtractor	5
16x1 Input multiplexer 5x2x2	20
2x16 Output multiplexer 5x2x2	20
Storage/latch	
12 bit latches 2x32x6	384
Clock and timing circuits	

```
C*************************************************************
C PROGRAM TO SIMULATE ANGULAR RESOLUTION USING A DFT ON
C SPATIAL SAMPLES
C*************************************************************
      REAL TAR(5,4),CLTR(5,4) !5 IS THE NUMBER OF TARGETS/CLUTTER
C                             !4 IS THE NUMBER OF ATTRIBUTES

      REAL TARX(5),TARY(5),TARZ(5) !THESE WILL GET THE DIRN COSINES
      REAL CLTRX(5),CLTRY(5),CLTRZ(5)
      REAL ANTX(32),ANTY(32),ANTZ(32),VECMAG(32)
      REAL ERRX,ERRY,ERRZ !THESE REPRESENT THE MAX ANT POS ERR ALONG
C           ANT PERPENDICULAR TO THE AXIS
      REAL PHASET(32,5),PHASEC(32,5)
      REAL JITTR
      REAL CAMP(32)
      INTEGER STOR(32),STOI(32),ROM1(32,16)
      INTEGER ROM2(32,16),ROM3(32,16),ROM4(32,16)
      INTEGER ARRSIZ,DCORIT,SAMPLS,MEMORY,POSMEM,KWANT,DATPOS
      INTEGER CLTR(32),OUTI(32)
      REAL ANG(16)
         COMPLEX U(32),W(256)
      REAL SF
      COMPLEX INSEQ(25,1),XCOM(25)
      REAL TMAG(32,25),ANAG(32),TSTMAG(32),AX(32)
      LOGICAL COLV,LIMTR,SER(32)
         REAL K,LAMBDA
      COMPLEX XY(32),TST(32)
         DIMENSION ARRAYS(32)
         DIMENSION ARRAY(32)
      DIMENSION AZ(32,25),ANP(32,25),FAZ(32,25),MA(25),SAZ(32)
         LOGICAL INIT,SORT,LIM*32,TAPER
      INTEGER IEX,NSTAG
      EQUIVALENCE (XCOM,INSEQ)
C DATA SECTION STARTS
      PIE=3.1415926535
      TWOPIE=2.*3.1415926535
      TWOPIT=TWOPIE
      DELTA=TWOPIE/360.
C     !NO. OF PRPS, NO. OF PLOTS EFFECTS THE DIMENSION OF
C     ARRAYS S,ARRAY, ARRAYS, AZ, ANP, FAZ, MA,INSEQ
C     ANG SUBROUTINE PLOT
C           ! NO. OF ANT. ELEMENTS EFFECTS DIMENSION OF ARRAYS
C   AZ,ANP, FAZ, MA, SUBROUTINES PLOT, A2D
C    ! D**IEX=NORM FOR FFT ROUTINE
C    !D=WAVELENGTH/FRAC 0 IS THE DISTANCE BETWEEN ANT
      NCLPR=5 !NO. OF CLUTTER POINTS CONSIDERED EFFECTS DIMENSION OF
```

```
C          ARRAY SEP, CRATIO
      NTAP=5
C DATA SECTION ENDS
C*****************************************************************
CINITIALIZES PARAMETERS FOR SIMULATION
C*****************************************************************
      SORT=.FALSE.
      READ 770,(SER(I),I=1,32)
770          FORMAT(16I1)
      DO 670 I=1,NCLTR
670          READ *,(CLTR(I,J),J=1,4)
      DO 671I=1,NTAP
671          READ *,(TAR(I,J),J=1,4)
      READ *,ANGLE,TOHOLD,ZM,N
      READ *,JITTR,ERRX,ERRY,ERRZ
      READ *,FRAC,LAMBDA,IEN
      READ *,CUTOFF
      READ 6001,LIMTR,LIMTR2,CONV,TAPER
6001          FORMAT(4L1)
      READ *,IORD,ISTAG,KWANT
      READ *,PROSTZ,SEANRO,POSNEW,DATPOS,OCORIT
C*****************************************************************
      SAMPLS=IORD;IEY=ISTAG
      D=LAMBDA/FRAC
      K=TURIE/LAMBDA
C*****************************************************************
CDETERMINES DIRECTION COSINES OF ANT. ELEMENTS,TARGETS &CLUTTER
C THETA1,ANGLE,  ANGLES ARE FROM BROADSIDE
C*****************************************************************
      DO 835 I=1,IORD
      AANTX=FLOAT(I-1)*LAMBDA/FRAC+ERRX*(RAN(DUM)-0.5)*2.
      AANTY=ERRY*(RAN(DUM)-0.5)*2.
      AANTZ=ERRZ*(RAN(DUM)-0.5)*2.
      VECMAG(I)=SQRT(AANTX**2+AANTY**2+AANTZ**2)
      IF(VECMAG(I).EQ.0.)ANTX(I)=0.
      IF(VECMAG(I).EQ.0.)ANTY(I)=0.
      IF(VECMAG(I).EQ.0.)ANTZ(I)=0.
      IF(VECMAG(I).EQ.0.)GO TO 835
      ANTX(I)=AANTY/VECMAG(I)
      ANTY(I)=AANTY/VECMAG(I)
      ANTZ(I)=AANTZ/VECMAG(I)
835          CONTINUE
      DO 825 I=1,NTAR
      THETA1=(ANGLE+TAR(I,1))*DELTA
      PL=TAR(I,2)*DELTA
```

```
      TARY(I)=COS(THETA1)*COS(EL)
      TARX(I)=SIN(THETA1)*COS(EL)
      TARZ(I)=SIN(EL)
825   CONTINUE
      DO 830 I=1,NCLTR
      THETA1=(ANGLE+CLTR(I,1))*DELTA
      EL=CLTR(I,2)*DELTA
      CLTRY(I)=COS(THETA1)*COS(EL)
      CLTRX(I)=SIN(THETA1)*COS(EL)
      CLTRZ(I)=SIN(EL)
830   CONTINUE
C*****************************************************************
C*****************************************************************
C CALCULATES COSINES OFANGLE BETWEEN POSITION VEC. OF TARGET/
C CLUTTER AND ANT. POS. VEC. & MULTIPLIES IT WITH THE ANT.
C POS. VECTOR VECMAG(I)
C*****************************************************************
      DO 810 IJ3=1,IORD
      DO 815 IJ4=1,NTAR
      PHASET(IJ3,IJ4)=K*(ANTX(IJ3)*TARX(IJ4)+ANTY(IJ3)*TARY(IJ4)
     1+ANTZ(IJ3)*TARZ(IJ4))*VECMAG(IJ3)
815   CONTINUE
      DO 820 IJ5=1,NCLTR
      PHASEC(IJ3,IJ5)=K*(ANTX(IJ3)*CLTRX(IJ5)+ANTY(IJ3)*CLTRY(IJ5)
     1+ANTZ(IJ3)*CLTRZ(IJ5))*VECMAG(IJ3)
820   CONTINUE
810   CONTINUE
C*****************************************************************
C*****************************************************************
C INITIALIZE TABLES FOR PASSING TO FFT SIMULATOR
C*****************************************************************
      IY=IORD/2;IX=2**DCOBIT
      DO 30 IJ=1,IY
      THETA=FLOAT(IJ-1)*PUPI/FLOAT(IORD)
      ANGL(IJ)=THETA*360./PUPI
      DO 35 IK=1,IX
      RCOS(IK,IJ)=IFIX((FLOAT(IK-1)/FLOAT(IX)*COS(THETA
     1     )*2**(BITPRC+0.5)*2**POSEXP
      RCOS(IK,IJ)=RCOS1(IK,IJ)
      RSIN(IK,IJ)=IFIX((FLOAT(-(IK-1))/FLOAT(IX)*SIN(THETA
     1     )*2**(BITPRC+0.5)*2**POSEXP
      RSIN(IK,IJ)=RSIN2(IK,IJ)
35    CONTINUE
30    CONTINUE
C     DO 31 IJ=1,IY
```

```
C   PRINT *,ANG(IJ),IJ
C   PRINT *,(RDM1(IK,IJ),RDM2(IK,IJ),IK,IK=1,IX)
31  CONTINUE
C*****************************************************************
      XA=3.142/FLOAT(IORD+1)
   N11=N-1
C*****************************************************************
C CALCULATE AZIMUTH ANGLES FOR PLOT POINTS
C*****************************************************************
   DO 23 IA=1,IORD
   IF(FRAC .GT.1.) GO TO 700
   X=FLOAT(IA-1)*FRAC/FLOAT(IORD)
   GO TO 710
700        CONTINUE
   X=FLOAT(IA-1)*FRAC/FLOAT(IORD)-(FRAC-1.)
710        AZ(IA,1)=ATAN(X/SQRT(1.-X**2))/DELTA
   SAZ(IA)=AZ(IA,1)
   DO 5555 IRA=1,N11
   AZ(IA,IRA+1)=AZ(IA,IRA)
 5555 CONTINUE
   23 CONTINUE
C*****************************************************************
C*****************************************************************
CCALCULATES PARAMETERS FOR PASSING TO RXR1, STMULATES RECETVER
C LIMITING AND COHERENT DETECTION
C*****************************************************************
   NW=1024;NW1=NW/4;DEL=TWPIE/FLOAT(NW);NW2=NW/2;NW3=NW2+NW1
   FREQ=300.*10.**6/LAMBDA;DT=1./(FREQ*FLOAT(NW))
   OMEGA=TWPIE*FREQ
   NW11=NW1+1
   DO 10 I=1,NW1;ST=SIN(FLOAT(I-1)*DEL);CO=COS(FLOAT(I-1)*DEL)
   W(I)=CMPLX(CO,ST)
10 CONTINUE
C*****************************************************************
C*****************************************************************
C NOISE, CLUTTER & SIGNAL PHASORS ADDED AT EACH ANT.,PASSED
C THRU LOGARITHIC RXR, & COHERENT DETECTION & A-D CONVERSION
C*****************************************************************
   IA=1
   PRINT 655,IA
655        FORMAT(6(/),10X,' COMPLEX NOISE SAMPLES AT ANTENNA',I4/)
      DO 21 IJ=1,N
   CALL NOISE(YY,IFX,IORD)
   RST(IJ)=YY(IA,IJ)
   PRINT 665,YY(IA,IJ)
```

```
656         FORMAT(40X,2E10.7)
      ANT=-PIE/2.
      DO 5 I=1,IORD
      ANT=ANT+XA
   RE=0.;AIM=0.
   DO 808 IJ1=1,NTAR
   TEMP=PHASET(I,IJ1)+TAR(IJ1,4)*FLOAT(IJ-1)
   RE=RE+COS(TEMP)*TAR(IJ1,3)
   AIM=AIM+SIN(TEMP)*TAR(IJ1,3)
808         CONTINUE
   DO 805 IJ2=1,NCLTR
   TEMP=PHASEC(I,IJ2)+CLTR(IJ2,4)*FLOAT(IJ-1)
   RE=RE+COS(TEMP)*CLTR(IJ2,3)
   AIM=AIM+SIN(TEMP)*CLTR(IJ2,3)
805         CONTINUE
      XX=SQRT(RE**2+AIM**2)
   PHASE=ATAN2(AIM,RE)
C*****************************************************************
C LIMITING RXR. & COHERENT IN-PHASE QUADRATURE DET.
C*****************************************************************
   IF(LIMTR)CALL RXROUT(XX,LAMBDA,PHASE,RE,AIM,CUTOFF)
   IF(LIMTR2)CALL RXR1(NW,NW1,DEL,NW2,NW3,N11,XX,OMEGA,PHASE,DT,X,
   1RE,AIM,CUTOFF,IITR)
   WT=1.
    IF(TAPER)     WT=0.54+0.46*COS(ANT)
   IF(.NOT.(SER(I))) WT=0.
      ARRAY(I)=WT*RE
      ARRAYS(I)=WT*AIM
    5 CONTINUE
C*****************************************************************
C A-D CONVERSION
C*****************************************************************
   IF(CONV ) CALL A2D(ARRAY,ARRAYS,STOR,STOI,IORD,KWANT,CUTOFF)
C*****************************************************************
C SPATIAL DFT
C*****************************************************************
   IF(CONV) CALL DFTSIM(KWANT,NPDSIZ,NEMERD,PDSMEM,DATPOS,DCOPIT,
   1IORD,NSMAG,OUTR,OUTI,ROW1,ROW2,ROW3,ROW4,STOR,STOI)
C  PRINT *,(OUTR(I),OUTI(I),I=1,IORD)
   IF(CONV) GO TO 1999
      DO 250 I1=1,IORD
      X(I1)=CMPLX(ARRAY(I1),ARRAYS(I1))
  250 CONTINUE
   CALL FFT(X,IDX,IORD)
1999        IF(.NOT.CONV)GO TO 2001
```

```
      DO 2000 I1=1,IORD;B(I1)=CMPLX(FLOAT(OUTR(I1)),FLOAT(OUTI(I1)))
2000  CONTINUE
2001  DO 252 I3=1,IORD
      AMP(I3,IJ)=0.
      FAZ(I3,IJ)=0.
      R=REAL(B(I3))
      A=AIMAG(B(I3))
      XL=R**2+A**2
      IF(XL .LE. THHOLD) GO TO 252
      AMP(I3,IJ)=SQRT(XL)
      FAZ(I3,I1)=ATAN2(A,R)/DELTA
252   CONTINUE
      MA(IJ)=IORD
C     PRINT 251,(AMP(I4,IJ),FAZ(I4,IJ),I4=1,IORD)
251   FORMAT(1H ,8F10.2)
      15 CONTINUE
      20 CONTINUE
      21 CONTINUE
C*****************************************************************
C*****************************************************************
      CALL FT(TST,TEN,N)
      DO 211 I=1,N
211   TSTMAG(I)=CABS(TST)
      PRINT 656,IANT,N
656   FORMAT(/40X,' NOISE SPECTRAL AMPLITUDE AT ANT. ',I4,' FOR ',
     1I4,' NOISE SAMPLES '/)
      PRINT 667,(TSTMAG(I),I=1,N)
667   FORMAT(40X,4F12.7)
      PRINT 771
771   FORMAT(/40X,'SERVICEABILITY STATE OF ANTENNAS')
      PRINT 772,(SER(I),I=1,32)
772   FORMAT(/40X,16L1)
      PRINT 601,ANGLE,THHOLD,ZN
601   FORMAT(40X,'TARGET AT ANGLE FROM BROADSIDE',F7.2,'DEG',3X,
     1 'THRESHOLD ',F7.2,' NOISE ',F7.2)
      PRINT 624,CUTOFF
624   FORMAT(40X,'CUTOFF = ',F7.2)
      PRINT 625,LISTR,LISTR2,COPY,TAPER
      PRINT 209,XPOSIZ  ,MEMORD,POSMEM,IORD,KVANT,DATPOS
209   FORMAT(40X,' XPOSIZ =',I5,' MEMORD=',I3,'POSMEM=',I3,'IORD=',
     1I3,'KVANT=',I3,'DATPOS=',I3)
      PRINT 649
      PRINT 650,((AMP(I,J),J=1,4),I=1,5)
      PRINT 651,((FAZ(I,J),J=1,4),I=1,5)
      PRINT 652,ERRX,ERRY,ERRZ
```

```
      PRINT 653,LAMBDA,FRAC
      PRINT 668
668           FORMAT(1H1,6(/))
649           FORMAT(40X,11X,'ANG. SEP.',1X,' ELEV. ',3X,' RATIO ',3X,' DOPPLE
     1R')
650           FORMAT(40X,'TARGET ',2X,4F10.5)
651           FORMAT(40X,'CLUTTER ',1X,4F10.5)
652           FORMAT(40X,'ERRY =',F10.2,'ERRY =',F10.2,'ERRZ =',F10.2)
653           FORMAT(40X,'LAMBDA =',F10.2,'FRAC =',F10.4 //)
      DO 661 I1=1,IORD
      CAMP(I1)=0.
      DO 660 I2=1,N
660           CAMP(I1)=CAMP(I1)+AMP(I1,I2)
      CAMP(I1)=CAMP(I1)/FLOAT(N)
661           CONTINUE
      CALL PLOT(SORT,1,SAZ,CAMP,MA,IORD)
      PRINT 668
C     CALL PLOT(SORT,N,AZ,FAZ,MA,IORD)
C*****************************************************************
C DIGITAL FILTER FOR MTI
C*****************************************************************
      DO 310 J=1,IORD
      DO 300 I=1,N
      TH=FAZ(J,I)*DELTA
      AA=AMP(J,I)*COS(TH);BB=AMP(J,I)*SIN(TH)
      INSEQ(I,1)=CMPLX(AA,BB)
300           CONTINUE
      CALL DIGFIL(INSEQ,N)
      TEM =0.
      DO 306 I=1,N
      TEM =TEM +CABS(INSEQ(I,1))
306           CONTINUE
      AMAG(J)=TEM/FLOAT(N)
310           CONTINUE
      PRINT 603,',ITER
603           FORMAT(//41X,'OUTPUT AFTER DOPPLER FILTERING , NO. OF SAMPLES ',I
     13,', & PHASE JITTER IS ',F7.2//)
605           FORMAT(10X,'ITER IS ',I3,',  LIMIT2 IS ',I3,',  CONV IS ',I3
     1,' TARGET IS ',I3)
      I=1
      J=1(1)=IORD
      PRINT 654
654           FORMAT(//40X,'---------- AZIL ----------->>>')
      CALL PLOT(SORT,1,SAZ,AMAG,MA,IORD)
22    CONTINUE
```

24 CONTINUE
STOP
END

```
10 A(J)=A(I)+1

20 U=U*

RETURN

END

SUBROUTINE A2D(ARRAY,ARRAYS,STOR,STOI,IORD,KWANT,CUTOFF)

DIMENSION ARRAY(IORD),ARRAYS(IORD)

INTEGER STOR(IORD),STOI(IORD)
DO 5 I=1,IORD

STOR(I)=IFIX(ARRAY(I)/CUTOFF*(2**KWANT-1))

STOI(I)=IFIX(ARRAYS(I)/CUTOFF*(2**KWANT-1))
 5 CONTINUE

RETURN

END
```

```
      SUBROUTINE FFTSIM(KWANT,WRDSIZ,MEMWRD,POSMEM,DATPOS,DCDBIT,IORD,
     1NSTAG,ISET,OUTI,ROM1,ROM2,ROM3,ROM4,STOR,STOI)
      INTEGER STOR(32),STOI(32),BTFLYR(16,2),BTFLYI(16,2),ROM1(32,16)
      INTEGER KOUNT(11)
      INTEGER ROM2(32,16),ROM3(32,16),ROM4(32,16)
      INTEGER WRDSIZ,DCDBIT,SAMPLS,MEMWRD,POSMEM,KWANT,DATPOS
      INTEGER ADIR11,ADIR12,ADII11,ADII12
      INTEGER ADIR21,ADIR22,ADII21,ADII22
      INTEGER ADDR1,ADDR2,ADDI1,ADDI2
      INTEGER ADRSR,ADRSI
      INTEGER REAL1,REAL2,IMAG1,IMAG2,REYUI,EMAG
      INTEGER X(5),OUTR(32),OUTI(32),MAG(32)
      COMPLEX XTEST(32)
      REAL ANG(16),XMAG(32)
      TUPI=2.*3.14159265358
C     TYPE 800
800   FORMAT(1X,'INPUT ISET ,IORD,NSTAG,KWANT')
C     ACCEPT *,ISET,IORD,NSTAG,KWANT
C     TYPE 208
208   FORMAT(1X,'INPUT WRDSIZ,MEMWRD,POSMEM,DATPOS,DCDBIT')
C     ACCEPT *,WRDSIZ,MEMWRD,POSMEM,DATPOS,DCDBIT
C     READ *,IRD
C     DO 004 IRD3=1,IRD
C     IORD=32;NSTAG=5;ISET=1
C     READ *,KWANT,WRDSIZ,MEMWRD,POSMEM,DATPOS,DCDBIT
      SAMPLS=IORD
      IY=IORD/2;IA=2**DCDBIT
C     DO 30 IJ=1,IY
C     THETA=FLOAT(IJ-1)*TUPI/FLOAT(IORD)
C     ANG(IJ)=THETA*360./TUPI
C           DO 35 IK=1,IX
C     ROM1(IK,IJ)=IFIX(FLOAT(IK-1)/FLOAT(IX)*COS(THETA
C     1     )*2**MEMWRD+0.5)*2**POSMEM
C           ROM3(IK,IJ)=ROM1(IK,IJ)
C     ROM2(IK,IJ)=IFIX(FLOAT(-(IK-1))/FLOAT(IX)*SIN(THETA
C     1     )*2**MEMWRD+0.5)*2**POSMEM
C           ROM4(IK,IJ)=ROM2(IK,IJ)
C  35             CONTINUE
C           CONTINUE
C     DO 31 IJ=1,IY
C     WRITE *,ANG(IJ),IJ
C     WRITE *,(ROM1(IK,IJ),ROM2(IK,IJ),IK,IK=1,IX)
C  31 CONTINUE
      IF (MOD(WRDSIZ,DCDBIT).NE.0) IA=IA+1;
      IX=WRDSIZ/DCDBIT;IY=2**DCDBIT;IZ=2**(WRDSIZ)
C     DO 004 IRD3=1,IRD
```

* IF (MOD(WRDSIZ, DCDBIT) .NE. 0) IA = IA + 1;

```
C   CALL SETPAR(ISET)
C   Z1=2**(NBITS-1)-1
    DO 700 I=1,IORD
C   XX=(2.*Z1*RAN(DUM)-Z1);YY=(2.*Z1*RAN(DUM)-Z1)
C   STOR(I)=IFIX(XX);STOI(I)=IFIX(YY);XTEST(I)=CMPLX(XX,YY)
    XTEST(I)=CMPLX(FLOAT(STOR(I)),FLOAT(STOI(I)))
700         CONTINUE
    DO 601 I=1,IORD;STOR(I)=STOR(I)*2**(DATPOS)
601         STOI(I)=STOI(I)*2**(DATPOS)
C600        TYPE 209,WRDSIZ  ,MEMWRD,POSMEM,IORD,KWANT,DATPOS
    DO 5 ISTAG=1,NSTAG
C   TYPE 101,(STOR(I),STOI(I),I=1,SAMPLS)
C   TYPE 201,ISTAG
201         FORMAT(1X,'ISTAG= ',I2)
101         FORMAT(1X,4I8)
    N1=ISTAG-1;N2=2**N1;N=2**(NSTAG-1)/N2
            DO 10 I1=1,N2
            N3=(I1-1)*N*2
501         CONTINUE
    DO 15 I2=1,N
    N4=N3+I2;N5=N4+N;IN=2**(ISTAG-1)*(I2-1)+1
C   TYPE 202,ISTAG,I1,I2,N4,N5,IN
202         FORMAT(1X,'ISTAG= ',I2,' I1= ',I3,' I2=',I3,' N4=',I3,'N5',I3
    1,' IN=',I3)
    BTFLYR(I2,1)=STOR(N4)+STOR(N5);BTFLYI(I2,1)=
    1STOI(N4)+STOI(N5)
    BTFLYR(I2,2)=STOR(N4)-STOR(N5);BTFLYI(I2,2)=
    1STOI(N4)-STOI(N5)
    ISGN1=1;ISGN1=ISIGN(ISGN1,BTFLYR(I2,2))
    ISGN2=1;ISGN2=ISIGN(ISGN2,BTFLYI(I2,2))
C   TYPE *,BTFLYR(I2,2),ISGN1,BTFLYI(I2,2),ISGN2
502         CONTINUE
C   TYPE 203,BTFLYR(I2,1),BTFLYI(I2,1),BTFLYR(I2,2),BTFLYI(I2,2)
203         FORMAT(1X,' BTFLYR(I2,1)=',I8,' BTFLYI(I2,1)=',I8,' BTFLYR(I2,2)
    1=',I8,' BTFLYI(I2,2)=',I8)
    IF(IABS(BTFLYR(I2,1)).GE.LIM.OR.IABS(BTFLYR(I2,2
    1)).GE.LIM.OR.IABS(BTFLYI(I2,1)).GE.LIM.OR.
    1IABS(BTFLYI(I2,2)).GE.LIM) GO TO 111
    STOR(N4)=BTFLYR(I2,1);STOI(N4)=BTFLYI(I2,1)
505         CONTINUE
    IF(ISTAG.EQ.NSTAG) GO TO 25
    STOR(N5)=BTFLYR(I2,2);STOI(N5)=BTFLYI(I2,2)
506         CONTINUE
    GO TO 15
25  ADIR11=0;ADIR12=0;ADIR21=0;ADIR22=0
```

```
      ADII11=0;ADII12=0;ADII21=0;ADII22=0
505   CONTINUE
      DO 20 I3=1,IA
      ADOR1=ADIR11+ADIR21;ADOR1=ADOR1/IX;ADIR21=ADOR1
      ADOR2=ADIR12+ADIR22;ADOR2=ADOR2/IX;ADIR22=ADOR2
      ADOI1=ADII11+ADII21;ADOI1=ADOI1/IX;ADII21=ADOI1
      ADOI2=ADII12+ADII22;ADOI2=ADOI2/IX;ADII22=ADOI2
C     TYPE 204,ADOR1,ADOR2,ADOI1,ADOI2
204   FORMAT(1X,' ADOR1=',I8,' ADOR2=',I8,' ADOI1=',I8,' ADOI2=',I8)
506   CONTINUE
      ITEM=IABS(BTFLYR(I2,2))/IX**(I3-1);ADRSR=ITEM-ITEM/IX*IX+1
      ITEM=IABS(BTFLYI(I2,2))/IX**(I3-1);ADRSI=ITEM-ITEM/IX*IX+1
C     TYPE 205,ADRSR,ADRSI,I3,IN
205   FORMAT(1X,' ADRSR=',I3,' ADRSI=',I3,' I3=',I3,' IN=',I3)
      ADIR11=ROM1(ADRSR,IN);ADIR12=ROM3(ADRSI,IN)
      ADII11=ROM2(ADRSR,IN);ADII12=ROM4(ADRSI,IN)
C     TYPE 206,ADIR11,ADIR12,ADII11,ADII12
206   FORMAT(1X,' ADIR11=',I8,' ADIR12=',I8,' ADII11=',I8,' ADII12=',
      1I8)
20    CONTINUE
      ADOR1=(ADIR11+ADIR21)*ISGN1;ADOR2=(ADIR12+ADIR22)*ISGN2;
      ADOI1=(ADII11+ADII21)*ISGN1
      ADOI2=(ADII12+ADII22)*ISGN2
      REYUL=ADOR1-ADOI2;EMAG=ADOI1+ADOR2
C     TYPE 207,REYUL,EMAG
207   FORMAT(1X,'REYUL=',I8,' EMAG=',I8)
507   CONTINUE
      IF(IABS(REYUL).GE.LIM.OR.IABS(EMAG).GE.LIM)
      1GO TO 111
      STOR(N5)=REYUL;STOI(N5)=EMAG
      GO TO 15
111   TYPE 100,ISTAG,I1,I2,N4,N5,STOR(N4),STOR(N5)
      1,STOI(N4),STOI(N5)
100   FORMAT(1X,'OVERFLOW OCCURRED. ISTAG =',I2,' I1=',I3,' I2=',I3,
      1' N4=',J3,' N5=',I3,' STOR(N4)=',I10,' STOR(N5)=',I10,' STOI(N4
      1)=',I10,' STOI(N5)=',I10)
      GO TO 803
15    CONTINUE
2     CONTINUE
5     CONTINUE
      DO 55 I1=1,IORD;DO 7 J=1,ISTAG
7     X(J)=0
      I=1;I=I1-1
4     ITEM1=I/2;IF(ITEM1 .EQ. 0) GO TO 6
      X(N)=COR(I,2);N=N+1;I=ITEM1;GO TO 4
```

```
      X(1)=1
      K=1;DO 605 K1=1,NSTAG;K11=K1-1;K=K+X(NSTAG-K11)*2**(K11)
605         CONTINUE
CC TYPE *,K,K11
      OUTR(K)=STOR(I1);OUTI(K)=STOI(I1)
      OUTR(K)=OUTR(K)/2**(DATPOS);OUTI(K)=OUTI(K)/2**(DATPOS)
C603        MAG(K)=SQRT(FLOAT(OUTR(K)**2+OUTI(K)**2))
55 CONTINUE
209         FORMAT(1X,' WRDSIZ =',I5,' MEMWRD=',I3,'POSMEM=',I3,'IORD=',
     1I3,'KWANT=',I3,'DATPOS=',I3)
      RETURN                                        !!!!!!!!!!!!!!!

C  TYPE *,(OUTR(K),OUTI(K),K,K=1,IORD)
C  CALL FT(XTEST,NSTAG,IORD)
CC TYPE *,(XTEST(I),I=1,IORD)
C  DO 56 I=1,IORD
C56         XMAG(I)=CABS(XTEST(I))
CC TYPE *,(XMAG(I),MAG(I),I,I=1,IORD)
C  CALL FTEST(XTEST,NSTAG,IORD)
C  SQRERR=0.;DO 58 I=1,11
C58         KOUNT(I)=0
C  DO 57 I=1,IORD
C  XMAG(I)=CABS(XTEST(I))
C  ERR=(XMAG(I)-FLOAT(MAG(I)))/XMAG(I)
C  IERR=IFIX(ABS(ERR)*100.)
C  IF(IERR.GE.10)GO TO 59
C  KOUNT(IERR+1)=KOUNT(IERR+1)+1
C  GO TO 57
C59         KOUNT(11)=KOUNT(11)+1
C  SQRERR=SQRERR+(ERR)**2
C57         CONTINUE
C  SIGERR=SQRT(SQREPR/FLOAT(IORD))
C  TYPE 802,SIGERR,IORD,(I,KOUNT(I),I=1,11)
C802        FORMAT(1X,'RMS DEVIATION=',F10.5,'NO. OF POINTS IN THE FFT',I3,
C  1(/1X,'NO. OF POINTS IN',I3,'IS',I3))
C  TYPE *,(XMAG(I),MAG(I),I,I=1,IORD)
      CALL SAVRAN(ISAV)
C  TYPE 801,ISAV
      ISAV=ISAV
888         CONTINUE
801         FORMAT(1X,'ISAV ='I12)
      RETURN
      END
```

```
      SUBROUTINE FTEST(X,IN,N)
      INTEGER IIX(10)
      COMPLEX X(N),W,BTFLY1,BTFLY2,TEM,TST
      TUPI=2.*3.14159;W=CMPLX(COS(TUPI/FLOAT(N)),-SIN(TUPI/FLOAT(N)))
      N2=N/2;NSTAG=IN
      DO 7 I2=1,NSTAG
      IY=2**(I2-1);DO 6 I3=1,IY
      IX=N2/IY;IZ=2*(I3-1)*IX+1;IZZ=IZ+(IX-1)
      DO 5 I1=IZ,IZZ
      I11=I1
      I1=I11+IX
C     TYPE *,X(I11),X(I1),I11,I1
      BTFLY1=X(I11)+X(I1);BTFLY2=X(I11)-X(I1)
      TST=W**((I11-IZ)*2**(I2-1))
20    X(I11)=BTFLY1;X(I1)=BTFLY2*TST
C     TYPE *,TST
125           CONTINUE
5     CONTINUE
6     CONTINUE
7     CONTINUE
      DO 13 I1=1,N
      DO 9 I4=1,NSTAG
9     IIX(I4)=0
      N1=1;I5=I1-1
11            ITEM=I5/2;IF(ITEM.EQ.0) GO TO 10
      IIX(N1)=MOD(I5,2);N1=N1+1;I5=ITEM;GO TO 11
10    IIX(N1)=I5
      K=1
      DO 12 KI=1,NSTAG;KI1=KI-1;NTEM=NSTAG-KI1;K=K+IIX(NTEM)*2**(KI1)
12    CONTINUE
C     TYPE *,K,I1
      IF(K .LE. I1) GO TO 13
      TEM=X(I1);X(K)=X(I1);X(I1)=TEM
13    CONTINUE
      TYPE *,I2
      TYPE *,(X(I1),I1=1,N)
      STOP ;END
```

```
      SUBROUTINE NOISE(XY,IN,N)
      REAL X(32,1),Y(32,1)
      INTEGER MA(1)
      COMPLEX XY(N)
C     COMPLEX XY(32),TEM(32)
C     N=32;IN=5
      TUPI=2.*3.14159
C     DO 20 IZ=1,32
      DO 5 I=1,N;FI=RAN(DUM)*TUPI;
      XY(I)=CMPLX(COS(FI),SIN(FI))
5     CONTINUE
      CALL FT(XY,IN,N)
      DO 10 I=1,N;XY(I)=CONJG(XY(I))/FLOAT(N)
10    CONTINUE
      RETURN
CC    TEM(IZ)=XY(16)
20    CONTINUE
C     DO 21 IZ1=1,N;XY(IZ1)=TEM(IZ1)
21    CONTINUE
CC    TYPE *,(XY(I),I=1,N)
C     CALL FT(XY,IN,N)
C     AVG=0.
C     DO 15 I=1,N;X(I,1)=CABS(XY(I));Y(I,1)=I
C     AVG=AVG+X(I,1)
15          CONTINUE
C     AVG=AVG/FLOAT(N)
C     SIGMA=0.;DO 25 I2=1,N;SIGMA=SIGMA+(X(I2,1)-AVG)**2
25    CONTINUE
C     SIGMA=SQRT(SIGMA)/FLOAT(N);TYPE *,AVG,SIGMA
C     MA(1)=N;SORT=.FALSE.;M=1
C     CALL PLOT(SORT,M,Y,X,MA,N)
CC    PAUSE
      END
```

```
      SUBROUTINE TEST1
      REAL LAMBDA
      COMPLEX W(256)
      LOGICAL LIMTR2
      TUPI=2.*3.14159
      TYPE 5
5     FORMAT(1X,'INPUT XIN,LAMBDA,PHASE,CUTOFF,ZM')
      ACCEPT *,XIN,LAMBDA,PHASE,CUTOFF,ZM
C     CALL RXROUT(XIN,LAMBDA,PHASE,OUTR,OUTI,CUTOFF)
      LIMTR2=.TRUE.
      NW=1024;NW1=NW/4;DEL=TUPI/FLOAT(NW);NW2=NW/2;NW3=NW2+NW1
      FREQ=300.*10.**6/LAMBDA;DT=1./(FREQ*FLOAT(NW))
      OMEGA=TUPI*FREQ
      NW11=NW1+1
      DO 10 I=1,NW1;SI=SIN(FLOAT(I-1)*DEL);CO=COS(FLOAT(I-1)*DEL)
      W(I)=CMPLX(CO,SI)
10    CONTINUE
      IF(LIMTR2)CALL RXR1(NW,NW1,DEL,NW2,NW3,NW11,XIN,OMEGA,PHASE,DT,W
      ,RE,AIM,CUTOFF,ZM)
      TYPE *,OUTR,OUTI,RE,AIM
      END
      SUBROUTINE RXR1(NW,NW1,DEL,NW2,NW3,NW11,XIN,OMEGA,PHASE,DT,W,RE,
      1AIM,CUTOFF,JITTR)
      REAL X(1024),XX(1024)
      REAL JITTR
C     COMPLEX Y(1024)
      COMPLEX W(NW1),TEMR,XY(1024)
      IEX=10
      TUPI=2.*3.14159
      PHASE=PHASE+JITTR*TUPI*RAN(DUM)
      CONTINUE
      DO 20 I=1,NW;TEMR=FLOAT(I-1)*DT
      X(I)=XIN*SIN(OMEGA*TEMR+PHASE
      X(I)=X(I)*COS(0.25)
      TYPE *,X(I),CUTOFF
      IF(ABS(X(I)).GT.CUTOFF)X(I)=CUTOFF*SIGN(1.,X(I))
C     TYPE *,X(I)
C     Y(I)=CMPLX(X(I),0.)
5     CONTINUE
C     TYPE *,(Y(I),I=1,...)
      RE=0.;AIM=0.;TEMR=(0.,0.)
C     GO TO 55
      DO 15 I=1,NW1
      TEMR=(X(I)-X(I+NW2))*CONJG(W(I))-W(NW11-I)*(X(NW1+I)-X(NW3+I))+
      ITEMR
```

```
      GO TO 15
20    TEMR=(X(I)-X(NW2+I))+(0.,-1.)*(X(NW1+I)-X(NW3+I))+TEMR
15          CONTINUE
      RE=REAL(TEMR);AIM=AIMAG(TEMR)
      RE=RE/FLOAT(NW)*2.;AIM=AIM/FLOAT(NW)*2.
C     TYPE *,RE,AIM
C55         CALL FT(Y,10,1024)
C     TYPE *,Y(2)
      RETURN
      END
```

```
      SUBROUTINE TEST4
      REAL LAMBDA
      COMPLEX W(250)
      LOGICAL LIMTR2
      TUPI=2.*3.14159
      TYPE 5
5     FORMAT(1X,'INPUT XIN,LAMBDA,PHASE,CUTOFF')
      ACCEPT *,XIN,LAMBDA,PHASE,CUTOFF
C     CALL RXROUT(XIN,LAMBDA,PHASE,OUTR,OUTI,CUTOFF)
      LIMTR2=.TRUE.
      NW=1000;NW1=NW/4;DEL=TUPI/FLOAT(NW);NW2=NW/2;NW3=NW2+NW1
      FREQ=300.*10.**6/LAMBDA;DT=1./(FREQ*FLOAT(NW))
      OMEGA=TUPI*FREQ
      NW11=NW1+1
      DO 10 I=1,NW1;SI=SIN(FLOAT(I-1)*DEL);CO=COS(FLOAT(I-1)*DEL)
      W(I)=CMPLX(CO,SI)
10    CONTINUE
      TYPE *,TUPI
      IF(LIMTR2)CALL RXR1(NW,NW1,DEL,NW2,NW3,NW11,XIN,OMEGA,PHASE,DT,W,RE
      1,RE,AIM,CUTOFF)
C     TYPE *,OUTR,OUTI
      STOP
      END
      SUBROUTINE RXROUT(XIN,LAMBDA,PHASE,OUTR,OUTI,CUTOFF)
      REAL LAMBDA,X(20,4),Y(4),YN(4),LIMIT,DT,COEFF(22)
C     REAL XD(100,1),YD(100,1) !CCC
      INTEGER MA(1)
      REAL XD(1),YD(1)
      LOGICAL SW
      DATA X/80*0./,Y/4*0./,YN/4*0./,COEFF/19*0.,1.,2*0./
C       M=100;N=1;MA(1)=100
      M=1;N=1
       START=0.
       SW=.FALSE.
       FREQ=300.*10.**6/LAMBDA
       T=1./FREQ
       OMEGA=2.*3.142*FREQ
       KNT1=1
       DT=T/1000.
       KNT1=1000
C      M=10
       CALL RXR(X,Y,YN,LIMIT,COEFF,DT,START,XD,YD,M,N,KNT1,SW,OMEGA,PHASE,)
      1OUT,COUTOFF)
C      CALL PLOT(.FALSE.,1,XD,YD,1A,5)
C      TYPE *,YD(100,1)*2./T
```

```
      OUTR=YD(1)*2./T
           AP=.TRUE.
C     TYPE *,OUTR
           CALL RK(X,Y,YN,LIMIT,COEFF,DT,START,XD,YD,M,N,KNT1,SW,OMEGA,
          1PHASE,XIN,CUTOFF)
C     CALL PLOT(.FALSE.,1,XD,YD,MA,M)
C          OUTI=YD(100,1)*2./T
      OUTI=YD(1)*2./T
           RETURN
           END
           SUBROUTINE RK(X,Y,YN,LIMIT,COEFF,DT,START,XD,YD,MZ,NQ,KNT1,SW,
          1OMEGA, PHASE,XIN,CUTOFF)
C FOURTH ORDER RUNGE-KUTTA FOR DIFF. EQ. UPTO ORDER 20
C COEFF(ZZ) ARE (COEFF(1).D**20+COEFF(2).D**19+.....+COEFF(20).D+COEFF(21))
C +COEFF(22) =FUNC(Y,T)
C D=DY/DT .
C X(I,1) ARE INITIAL VALUES OF D**19 TO D
C Y(1) INITIAL VALUE OF Y
C X(I,I),X(I,0),X(I,J),Y(2),Y(3),Y(4) ARE INTERMEDIATE VALUES WHILE CALCULATIN
C 4TH ORDER RUNGE-KUTTA
C XD(I) ARE VALUES OF D**20 CALCULATED
C START  SPECIFIES THE INITIAL TIME
C LIMIT IS FINAL TIME
C DT IS TIME INCREMENT
C APLOT SHOULD BE SET TO TRUE IF APLOT IS WANTED ELSE TO FALSE
C M IS THE NUMBER OF POINTS PLOTTED AND IS THE DIMENSION OF XD,YD.
C NOTE M=(LIMIT-START)/(DT*KNT1)
C ARRAYS XD,YD RETURN THE RESULTS
C A REAL FUNCTION SUBPROGRAM MUST BE PROVIDED
           REAL COEFF(22),COEF(22)
           REAL X(20 , 4) , Y(4) , YN(4) , LIMIT,Y1(4)
C          DIMENSION XD(MZ,NQ),YD(MZ,NQ)
      DIMENSION XD(1),YD(1)
           DIMENSION  SLOPE(4)
           DO 2000 I1=1,4
            Y(I1)=0.
      Y1(I1)=Y(I1)
      DO 2001 I1=1,22;COEF(I1)=COEFF(I1)
2001       CONTINUE
           DO 15 I=1,20
      I1=I
      IF (COEF(I) .NE. 0.) GO TO 20
           STOP
      20 I=21- I1
```

```
      IF(II .EQ. 1) GO TO 30
       N=1
   25 COEF(N) = COEF(II)
      IF(II .GT. 19) GO TO 26
      X(N,1)=X(II,1)
   26 N=N+1
      II=II+1
       IF(II .LE. 22) GO TO 25
   30 M=J+2
        DO 35 K=2,M
   35 COEF(K) = COEF(K)/COEF(1)
      CONST = 1./COEF(1)
       COEF(1)=1.0
C      TYPE 500,(COEF(I),I=1,M)
C      TYPE 31,Y1(1),(X(I-1,1),I=2,J)
   31 FORMAT(1H ,'INITIAL CONDITION FIRST IS Y1(0) NEXT FROM RIGHT TO LEFT
     1T D(0),D(0)**2----',(/1H ,10F10.2))
  500 FORMAT(1H ,'COEF ARE IN DECREASING POWERS OF D RIGHT TO LEFT. RIGHTMOS
     1HTMOST IS CONST. NEXT D**0----',(/1H ,5F10.2))
       KK=0
      STEP=START
      J=J-1
      TIME = 0.
  120 TERM = 0.
      I = STEP
      IF(I .NE. K(1)) GO TO 123
      K=2
      KK=KK+1
C      XD(KK,1)=TIME
C      YD(KK,1)=Y1(1)
   XD(KK)=TIME;YD(KK)=Y1(1)
  123 CONTINUE
       IF(STEP .GT. LIMIT) RETURN
      IF(M .EQ. 0) GO TO 41
        DO 40 I=1,M
   40 TERM = COEF(I+1)*X(I,1) + TERM
   41 Y1(1)=CONST*FUNC(TIME,XIN,OMEGA,PHASE,SW,CUTOFF)-TERM-COEF(J+1)
     1*Y1(1)=COEF(J+2)
      WRITE(14,'D(1) IS ',E14.4)
       SLOPE = Y1(1)
      IF(J .EQ. 0) GO TO 51
      DO 50 I=1,J
      X(I,2)=SLOPE*DT/2.+X(I,1)
   50 SLOPE=X(I,1)
   51 Y1(2)=Y1(1)+SLOPE*DT/2.
```

```
      TIME=DT/2.+TIME
       TERM = 0.
      IF(M .EQ. 0) GO TO 61
      DO 60 I=1,M
   60 TERM=COEF(I+1)*X(I,2)+TERM
61    YN(2)=CONST*FUNC(TIME,XIN,OMEGA,PHASE,SW,CUTOFF)-TERM-COEF(J+1)
     1*Y1(2)-COEF(J+2)
  502 FORMAT(1H ,'X(I,I) IS ',4E14.4)
       SLOP=YN(2)
      IF(M .EQ. 0) GO TO 71
       DO 70 I=1,M
      X(I,3)= SLOP*DT/2. + X(I,1)
   70 SLOP=X(I,2)
71     Y1(3)=Y1(1) + SLOP*DT/2.
      TERM=0.
      IF(M .EQ. 0) GO TO 81
      DO 80 I=1,M
   80 TERM = TERM+ COEF(I+1)*X(I,3)
81    YN(3)=CONST*FUNC(TIME,XIN,OMEGA,PHASE,SW,CUTOFF)-TERM-COEF(J+1)
     1*Y1(3)-COEF(J+2)
  503 FORMAT(1H ,'X(I,3) IS ',4E14.4)
      TIME=DT/2.+TIME
      SLOP=YN(3)
      IF(M .EQ. 0) GO TO 91
      DO 90 I=1,M
      X(I,4)=SLOP*DT+X(I,1)
   90 SLOP=X(I,3)
91    Y1(4)=Y1(1)+SLOP*DT
      TERM =0.
      IF(M .EQ. 0) GO TO 101
      DO 100 I=1,M
  100 TERM =TERM+COEF(I+1)*X(I,4)
101   YN(4)=CONST*FUNC(TIME,XIN,OMEGA,PHASE,SW,CUTOFF)-TERM-COEF(J+1)
     1*Y1(4)-COEF(J+2)
  504 FORMAT(1H ,'X(I,4) IS ',4E14.4)
      DO 105 L=1,4
  105  SLOPE(L) = YN(L)
      IF(M .EQ. 0) GO TO 111
      DO 110 I=1,M
      AVSLOP=(SLOPE(1)+SLOPE(2)*2.+2.*SLOPE(3)+SLOPE(4))*1./6.*DT
     1+X(I,1)
      DO 110 L=1,4
  110 SLOPE(L)=X(I,L)
  111 X(I,1) =AVSLOP
111   Y1(1)= Y1(1)+(SLOPE(1)+2.*SLOPE(2)+2.*SLOPE(3)+SLOPE(4))/6.*DT +Y1(1)
```

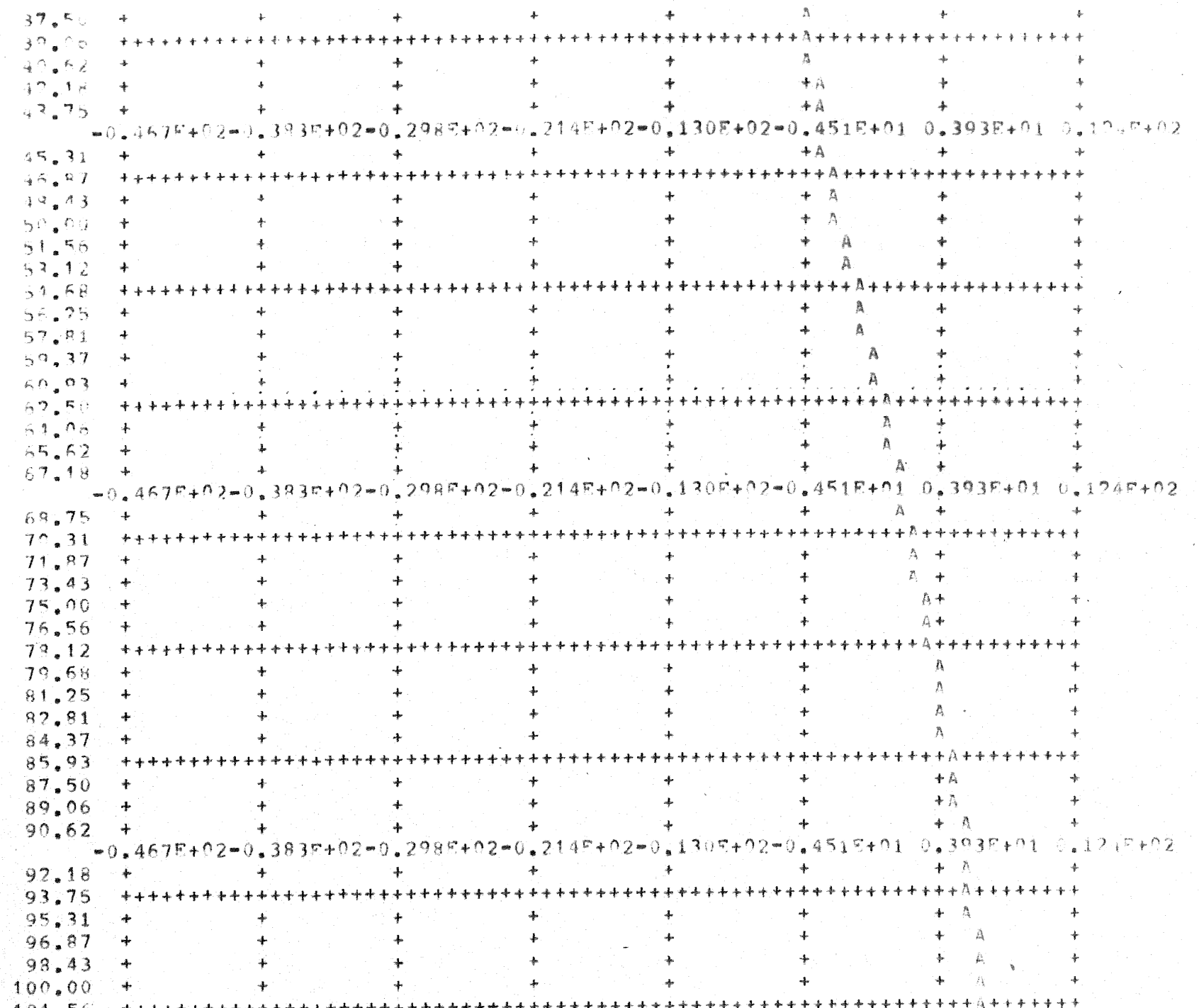
```
1001  STEP=STEP+DT
       KNT=KNT+1
   GO TO  120
      RETURN
      END
      REAL FUNCTION FUNC(TIME,XIN,OMEGA,PHASE,SW,CUTOFF)
      ANGLE=OMEGA*TIME
      X=XIN*COS(ANGLE+PHASE)
      IF(ABS(X) .GE. CUTOFF)X=CUTOFF*SIGN(1.,X)
      IF(SW) GO TO 5
      FUNC=X*COS(ANGLE)
      RETURN
    5 FUNC=X*SIN(ANGLE)
      RETURN
      END
```

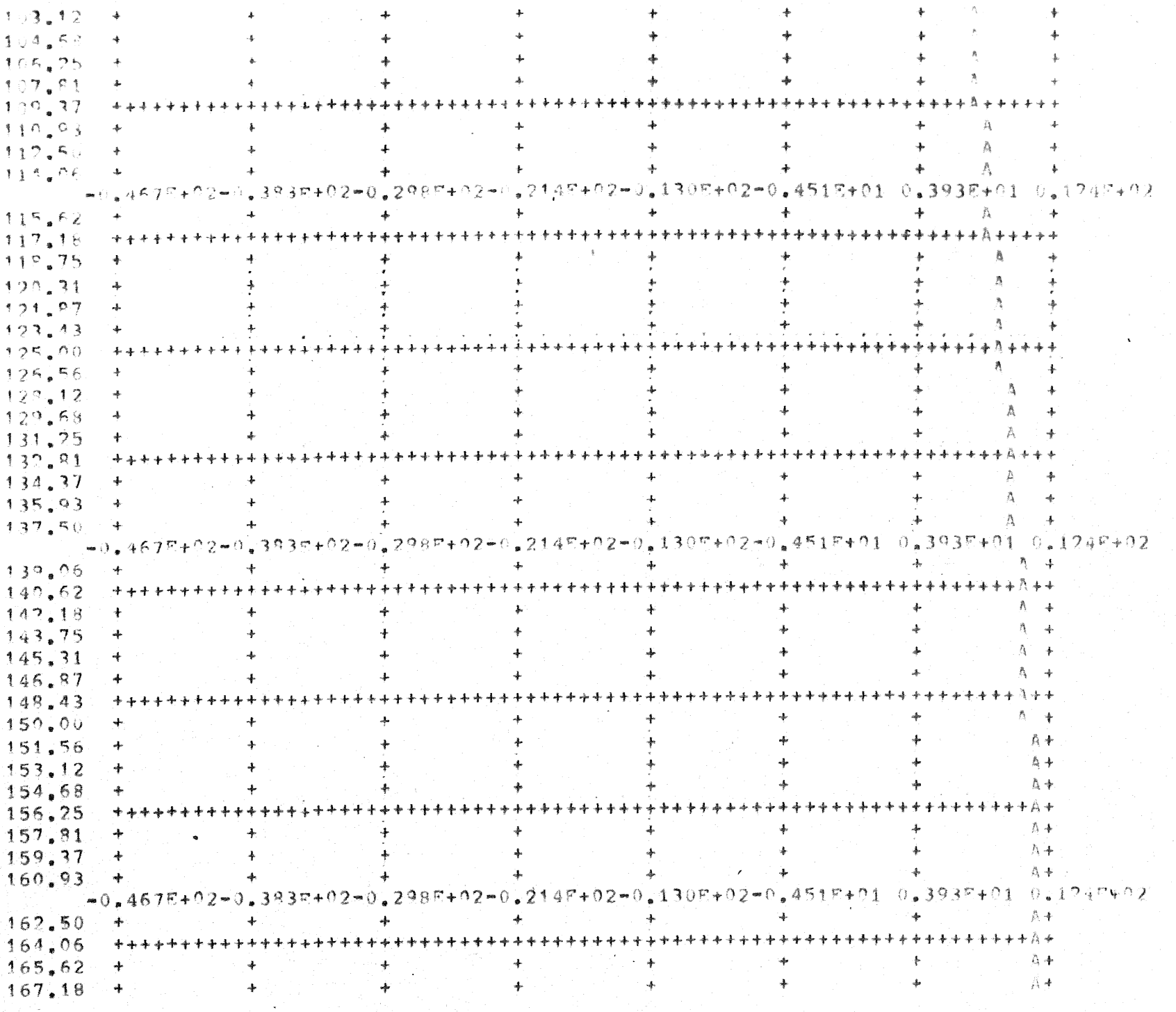
```
C   SUBROUTINE TEST11
    REAL INSEQ(256,1),FREQ(256,1),SPMAG(256,1)
    DIMENSION MA(1)
    COMPLEX SPEC(256)
    LOGICAL SORT
    SORT=.FALSE.
    I=1
    TUPI=3.14159*2.
    PRT=1./400.
    PHI=30.*TUPI/360.0
    N=256
    IN=8
    DO 10 J=1,N
    FREQ(J,1)=FLOAT(J-1)*400./FLOAT(N)
C   FRE=FLOAT(J-1)*6.
10  CONTINUE
    INSEQ(1,1)=1.
    DO 5 I=2,N
C   TIM=FLOAT(I-1)*PRT
C   INSEQ(1,1)=SIN(TUPI*FRE*TIM+PHI)
    INSEQ(I,1)=0.
  5 CONTINUE
C   TYPE *,(INSEQ(K,1),K=1,N)
    CALL DIGFIL(INSEQ,N)
C   TYPE *,(INSEQ(K,1),K=1,N)
C   TYPE *,TUPI
C   TYPE *,PHI
C10         CONTINUE
    DO 15 I=1,N
    SPEC(I)=CMPLX(INSEQ(I,1),0.)
15  CONTINUE
    CALL FT(SPEC,IN,N)
    DO 20 I=1,N
    SPMAG(I,1)=10.*ALOG10(CABS(SPEC(I)))
20  CONTINUE
    MA(1)=256
    PRINT 900
900         FORMAT(//7X,'----------------IN DBS ------------->')
    CALL PLOT(SORT,M,FREQ,SPMAG,MA,N)
    STOP
    END
    SUBROUTINE DIGFIL(INSEQ,N)

    REAL DNOM1(3),DNOM2(3),DNOM3(3),DNOM1(3),DNOM2(3),DNOM3(3),K
C   COMPLEX INSEQ(N,1)
C   DIMENSION INSEQ(7,256)
```

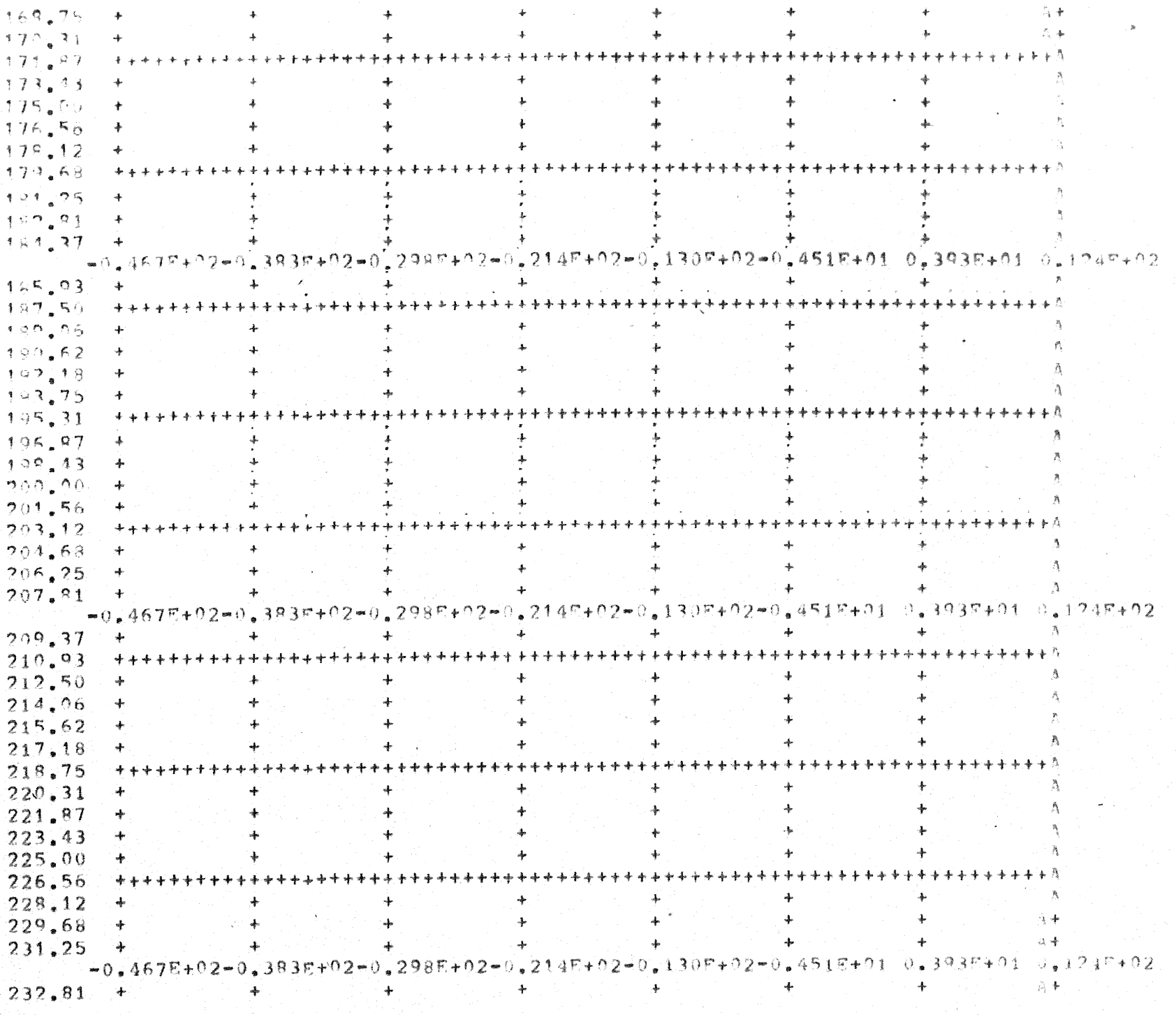
```
      REAL INSEQ(N,1),INTER(7,256)
      DATA NUM1/1., 1.,1./,NUM2/-1.99,-1.95,-1.999/,NUM3/3*1./
      DATA DNOM1/3*1./,DNOM2/0.,-1.7,0./,DNOM3/0.,0.95,0./
      DATA A/1./
C     READ *,(NUM1(I),NUM2(I),NUM3(I),I=1,3)
C     READ *,(DNOM1(I),DNOM2(I),DNOM3(I),I=1,3)
      PRINT 901
      PRINT 903,(NUM1(I),NUM2(I),NUM3(I),I=1,3)
901           FORMAT(6(/)7X,'NUMERATOR COEFF')
900           FORMAT(///7X,'DENOMINATOR COEFF')
      PRINT 900
903           FORMAT(7X,3F7.4)
      PRINT 903,(DNOM1(I),DNOM2(I),DNOM3(I),I=1,3)
      TWPI=2.*3.14159
      DO 59 IM=1,N
      INTER(1,IM)=INSEQ(IM,1)
59    CONTINUE
      DO 13 I=2,6,2
      L=I/2
      INTER(I,1)=INTER(I-1,1)*DNOM1(L)
      INTER(I+1,1)=NUM1(L)*INTER(I,1)
13            CONTINUE
      DO 131 I1=2,6,2
      L=I1/2
      INTER(I1,2)=DNOM1(L)*(INTER(I1-1,2)-DNOM2(L)*INTER(I1,1))
      INTER(I1+1,2)=NUM1(L)*INTER(I1,2)+NUM2(L)*INTER(I1,1)
131           CONTINUE
      DO 15 K=3,N
      DO 14 J=2,6,2
      I=J/2
      INTER(J,K)=DNOM1(L)*(INTER(J-1,K)-DNOM2(I)*INTER(J,K-1)-DNOM3(I)
     1*INTER(J,K-2))
      INTER(J+1,K)=INTER(J,K)*NUM1(I)+INTER(J,K-1)*NUM2(I)+INTER(J,K-2
     1)*NUM3(I)
14    CONTINUE
15    CONTINUE
      DO 16 M=1,N
      INSEQ(M,1)=INTER(7,M)*A
C     IF(M .LE.4) INSEQ(M,1)=(FLOAT(M-1)/5.)*INSEQ(M,1)
16    CONTINUE
      RETURN
      END
```

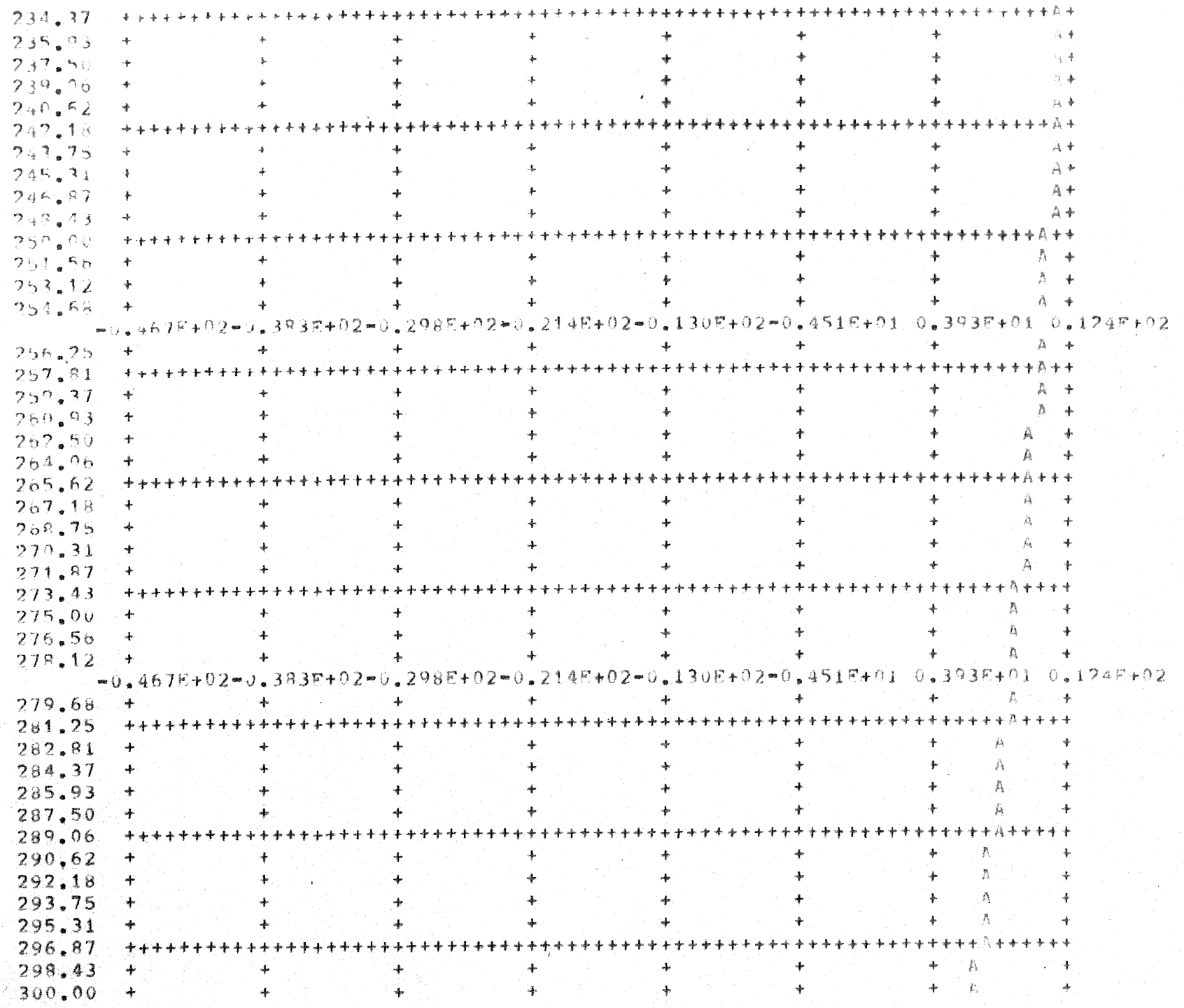
NUMERATOR COEFF
1.0000-1.0000 1.0000
1.0000-1.9500 1.0000
1.0000-1.9000 1.0000

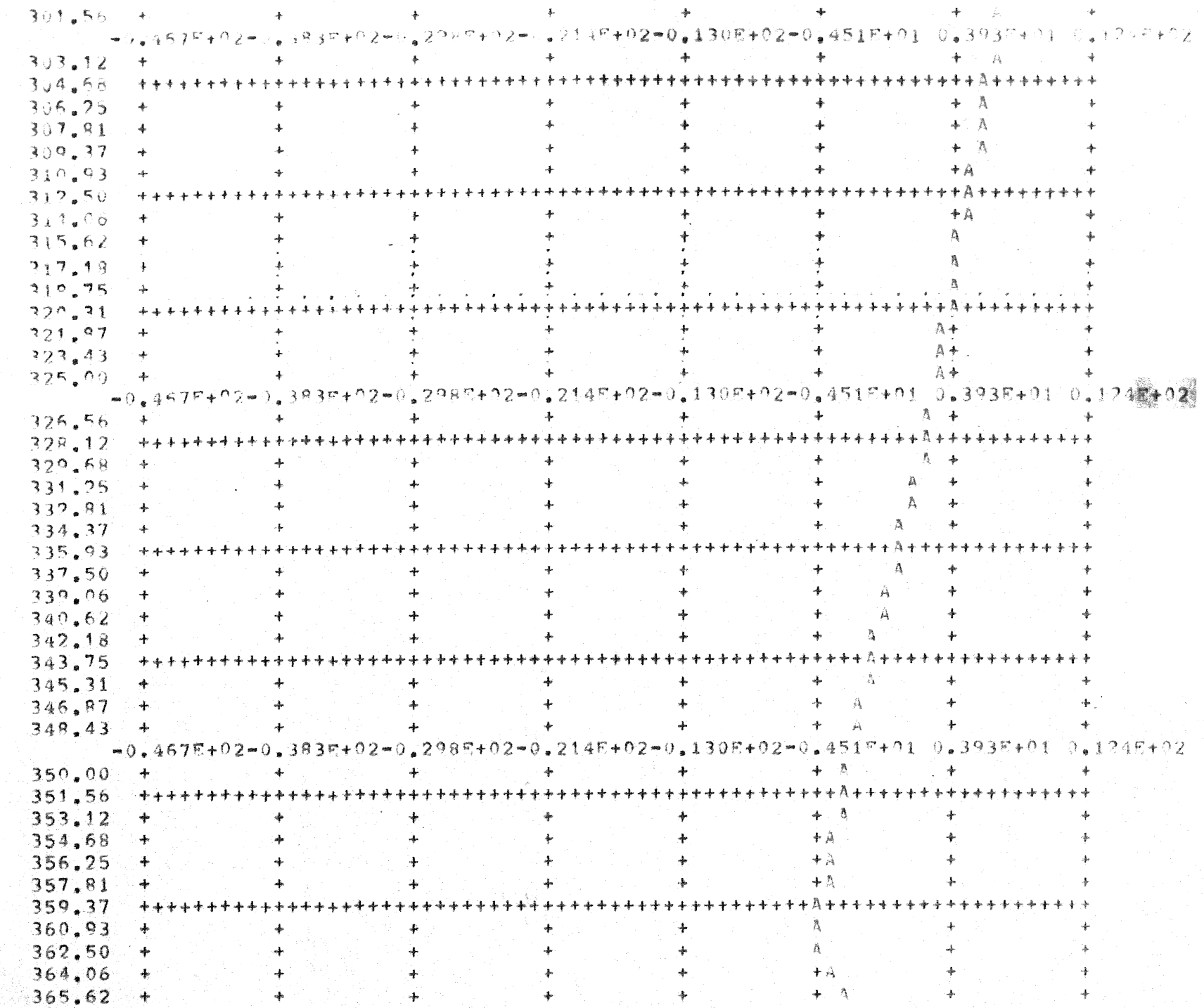
DENOMINATOR COEFF
1.0000 0.0000 0.0000
1.0000-1.7000 0.9500
1.0000 0.0000 0.0000


-0.467E+02 -0.383E+02 -0.298E+02 -0.214E+02 -0.130E+02 -0.451E+01 0.393E+01 0.124E+02
-0.467E+02 -0.383E+02 -0.298E+02 -0.214E+02 -0.130E+02 -0.451E+01 0.393E+01 0.124E+02
-0.467E+02 -0.383E+02 -0.298E+02 -0.214E+02 -0.130E+02 -0.451E+01 0.393E+01 0.124E+02


-0.467E+02 -0.383E+02 -0.298E+02 -0.214E+02 -0.130E+02 -0.451E+01 0.393E+01 0.124E+02
-0.467E+02 -0.383E+02 -0.298E+02 -0.214E+02 -0.130E+02 -0.451E+01 0.393E+01 0.124E+02
-0.467E+02 -0.383E+02 -0.298E+02 -0.214E+02 -0.130E+02 -0.451E+01 0.393E+01 0.124E+02


168.75
170.31
171.87
173.43
175.00
176.56
178.12
179.68
181.25
182.81
184.37
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
185.93
187.50
189.06
190.62
192.18
193.75
195.31
196.87
198.43
200.00
201.56
203.12
204.68
206.25
207.81
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
209.37
210.93
212.50
214.06
215.62
217.18
218.75
220.31
221.87
223.43
225.00
226.56
228.12
229.68
231.25
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
232.81


234.37
235.93
237.50
239.06
240.62
242.18
243.75
245.31
246.87
248.43
250.00
251.56
253.12
254.68
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
256.25
257.81
259.37
260.93
262.50
264.06
265.62
267.18
268.75
270.31
271.87
273.43
275.00
276.56
278.12
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
279.68
281.25
282.81
284.37
285.93
287.50
289.06
290.62
292.18
293.75
295.31
296.87
298.43
300.00


301.56
303.12
304.68
306.25
307.81
309.37
310.93
312.50
314.06
315.62
317.18
318.75
320.31
321.87
323.43
325.00
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
326.56
328.12
329.68
331.25
332.81
334.37
335.93
337.50
339.06
340.62
342.18
343.75
345.31
346.87
348.43
-0.467E+02-0.383E+02-0.298E+02-0.214E+02-0.130E+02-0.451E+01 0.393E+01 0.124E+02
350.00
351.56
353.12
354.68
356.25
357.81
359.37
360.93
362.50
364.06
365.62